VERIFICATION AND FEASIBILITY STUDY OF A MICRO-COMPUTER BASED BALLISTICS ALGORITHM

John Thomas Ertlschweiger

DUDLEY KNOX LIBRARY NAVAL POSTGRADUATE SCHOOL

NAVAL POSTGRADUATE SCHOOL

Monterey, California



THESIS

VERIFICATION AND FEASIBILITY STUDY
OF A MICRO-COMPUTER BASED
BALLISTICS ALGORITHM

by

John Thomas Ertlschweiger II

December 1976

Thesis Advisor:

U. R. Kodres

Approved for public release; distribution unlimited.

7170321



REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM	
I. REPORT NUMBER	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER	
Verification and Feasibility Study of A Micro-Computer Based Ballistics Algorithm		5. Type of Report & Period Covered Master's Thesis December 1976 6. Performing org. Report Number	
John Thomas Ertlschweiger I	I	8. CONTRACT OR GRANT NUMBER(*)	
Naval Postgraduate School Monterey, California 93940		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS	
Naval Postgraduate School Monterey, California 93940		December 1976 13. NUMBER OF PAGES 247	
Naval Postgraduate School Monterey, California 93940		Unclassified 15a. Declassification/Downgrading Schedule	

Approved for public release; distribution unlimited.

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse eide if necessary and identify by block number)

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

The radical cost reductions in computer hardware brought about by large scale integration (LSI) has motivated this feasibility study which explores the use of the INTEL 8080 as a ballistics computer in a distributed micro-computer based airborne tactical weapons system.

The results show that software floating point arithmetic



20. Abstract (Cont'd)

using a sixteen bit mantissa is sufficiently accurate for solving the ballistics problem.

Experimental data failed to show that the mathematical model accurately predicts the weapon's behavior. Either the instrumentation to record the release data was inaccurate, or the ballistics tables do not accurately predict the actual behavior of falling weapons.



Verification and Feasibility Study of a Micro-Computer Based Ballistics Algorithm

by

John Thomas Ertlschweiger II Lieutenant, United States Navy B.A., University of Virginia, 1969

Submitted in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE IN COMPUTER SCIENCE

from the

NAVAL POSTGRADUATE SCHOOL December 1976 1 hesis 2654 c.1

ABSTRACT

The radical cost reductions in computer hardware brought about by large scale integration (LSI) has motivated this feasibility study which explores the use of the INTEL 8080 as a ballistics computer in a distributed micro-computer based airborne tactical weapons system.

The results show that software floating point arithmetic using a sixteen bit mantissa is sufficiently accurate for solving the ballistics problem.

Experimental data failed to show that the mathematical model accurately predicts the weapon's behavior. Either the instrumentation to record the release data was inaccurate, or the ballistics tables do not accurately predict the actual behavior of falling weapons.



TABLE OF CONTENTS

I.	INT	RODUCTION	10
	Α.	FEASIBILITY STUDY	11
	В.	VERIFICATION	11
	C.	PRESENTATION OF THE THESIS	12
II.	BAC	KGROUND	14
	Α.	THE NAVIGATION COMPUTER	14
	В.	THE BALLISTICS COMPUTER	16
	C.	THE EXECUTIVE COMPUTER	16
	D.	THE COMMUNICATIONS SCHEME	17
III.	BAL	LISTICS PROBLEM	19
	Α.	THE MATHEMATICAL MODEL	19
	В.	BASIC ASSUMPTIONS	20
	C.	DERIVATION OF DOWN RANGE TRAVEL	21
IV.		CO-COMPUTERS IN AIRBORNE TACTICAL PON SYSTEMS	26
	Α.	THE QUESTION OF ACCURACY	26
	В.	THE QUESTION OF SPEED	29
V.	VER	IFICATION OF DOWN RANGE TRAVEL	32
	Α.	THE BALLISTICS ALGORITHM - FORTRAN VS. PLM	32
	В.	OBSERVED DATA VS. BALLISTICS TABLES	33
		1. Predicted Down Range Travel	33



2. Wind Effect	36
3. Observed Slant Range	40
a. Method for DRT Calculation	40
b. Error Analysis	41
4. Hit Distances	43
VI. PRESENTATION OF RESULTS	44
A. FORTRAN VS. PLM	44
B: BALLISTICS TABLES VS. FORTRAN RESULTS	46
C. EXPERIMENTAL DATA VS. FORTRAN ALGORITHM -	48
VII. CONCLUSION	50
APPENDIX A	52
APPENDIX B 1	11
APPENDIX C 1	70
APPENDIX D 1	82
APPENDIX E 1	97
FORTRAN VERSION of the BALLISTICS ALGORITHM 2	10
PLM VERSION of the BALLISTICS ALGORITHM 2	23
LIST OF REFERENCES 2	46
INITIAL DISTRIBUTION LIST 2	47



LIST OF TABLES

I	AMOUNT OF TIME BALLISTICS PROGRAM SPENDS IN THE FLOATING POINT PACKAGE 3:
II	SUMMARY OF FORTRAN VS. PLM RESULTS 4
III	SUMMARY OF BALLISTICS TABLES VS. FORTRAN RESULTS 4
IV	SUMMARY OF BALLISTICS TABLES VS. ACTUAL DELIVERY DATA 49



LIST OF FIGURES

1.	ORGANIZATION OF MULTIPLE PROCESSOR AIRBORNE TACTICAL WEAPONS SYSTEM	18
2.	EFFECTS OF BALLISTICS TABLES ERROR SENSITIVITIES FOR HEIGHT	35
3.	DESIRED ERROR SENSITIVITIES FOR BALLISTICS PROBLEM	35
4.	NO WIND SOLUTION TO DOWN RANGE TRAVEL	39
5.	WIND CORRECTED SOLUTION TO DOWN RANGE TRAVEL	39



ACKNOWLEDGEMENTS

The author wishes to express his sincere gratitude and thanks to Lieutenant Gordon Eubanks, USN, whose technical skill and expertise in the field of microcomputer programming and hardware trouble shooting proved to be an invaluable aid in compiling the data presented in the appendices.

Also special thanks are due to Kathryn B. Strutynski for her help in developing a software package to convert the three byte floating point data into IBM's four byte floating point format.

Last but certainly not least, thanks go to my wife, Beverlea, and children Kimberly and Tom, who persevered many lonely weekends and nights, and who provided the encouragement and inspiration required throughout the preparation of this document.



I. INTRODUCTION

Military airborne tactical weapon systems have been designed and implemented primarily to aid the aircrew in performing their mission with accuracy and speed. This usually means that a shorter time is spent over target which increases the survivability of both the weapon systems platform and the aircrew.

The system presently employed on board the A6E, one of the Navy's attack aircraft, utilizes an IBM 4 PI series mini-computer to perform two major functions.

- 1. Navigation.
- 2. Solution of the ballistics problem.

In addition, several other related functions are performed by the system.

- 3. It provides steering commands.
- 4. It provides real-time display of sensor information.
- 5. It provides release pulses to the weapon at the appropriate time.

A. FEASIBILITY STUDY

This thesis will attempt to prove the feasibility of implementing an airborne tactical weapons system using



micro-computers. Two important questions must be answered in order to establish the feasibility of using a micro-computer in a tactical weapons system:

- 1. Is the micro-computer accurate enough?
- 2. Is the micro-computer fast enough?

The accuracy problem was approached by executing the ballistics algorithm on an IBM system 360 using a 21 to 24 bit mantissa in the standard floating point format and comparing the results of the same algorithm executed on an INTEL 8080 micro-processor with a 16 bit floating point mantissa. The question of speed was answered by executing the ballistics algorithm for numerous weapon types and initial conditions and observing the elapsed clock time.

B. VERIFICATION

The second aspect of this thesis was to verify that the ballistics algorithm corresponded to published tables as well as experimental data. The NAVAIR 01-1C-1T-1 ballistics tables were used to compare time of fall and down range travel against the results of the FORTRAN version of the ballistics algorithm. A total of 1813 different initial release conditions were examined spanning 18 different weapon types, various dive angles from +10 degrees to -60 degrees, altitudes from 500 feet to 15,000 feet, and air-speeds from 300 knots to 650 knots.



The source of experimental data was a set of data cards recorded by the bombardiers of an A-6E squadron over a period of one year. The data for each bomb drop consists of 24 various computer readouts at the instant the weapon is released from the aircraft as well as the hit coordinates of the weapon. This information is used to determine the initial conditions for the ballistics equations. Unfortunately a critical parameter, the dive angle, was recorded only to the nearest degree. Consequently an error analysis was conducted to determine the maximum error which could be expected from the rounding of the output data.

C. PRESENTATION OF THE THESIS

Chapter II explains the organization of a distributed micro-computer airborne tactical weapons system and discusses how the output of each subsystem is integrated with the entire system. The ballistics problem and the derivation of the differential equations used to describe the mathematical model are discussed in detail in Chapter III. Since no analytical solution exists for these equations, a simplified version of the model is solved analytically to gain insight.

Chapter IV explores the feasibility of using microprocessors in an airborne tactical weapons system.



Aspects of accuracy and speed are examined. Factors in attempting verification of the ballistics algorithm and experimental data with the NAVAIR 01-1C-1T-1 ballistics tables are contained in Chapter V.

Chapter VI presents the results of this thesis. The conclusions and recommendations of the author concerning this thesis are written in Chapter VII.

Appendix A contains the output from the FORTRAN program comparing the FORTRAN and PLM versions of the ballistics algorithm. Appendix B presents the same comparison between the FORTRAN version of the ballistics algorithm and the ballistics tables. Appendix C is a listing of the experimental data as they were recorded from the cockpit readouts of various A-6E aircraft. Appendices D and E are the results of the experimental delivery data compared against the FORTRAN version of the ballistics algorithm (approximating the ballistics tables), using two sets of drag coefficients.

The ballistics algorithm used in this thesis is the SIGMA version of the BOEING ALGORITHM modified by the Naval Weapons Laboratory at China Lake, California. The algorithm was further modified at the Naval Postgraduate School in Monterey, California for eventual implementation on the ballistics processor of the multiple micro-processor tactical weapons system.



II. BACKGROUND

To provide a better understanding of the role the ballistics processor plays in the multiple processor system, a brief overview of the proposed system will be discussed. The computer system is composed of three microprocessors: a navigation computer, a ballistics computer, and an executive computer. Each machine is dedicated to the process to which it is assigned instead of sharing resources of a single processor as in the present operational systems.

A. THE NAVIGATION COMPUTER

The navigation computer is a basic element to all tactical systems. In present operational systems, the navigation program is executed periodically to update the present position by the change in position since the last time increment. The navigation computer utilizes input from four major sensor instruments as its primary source of information.

- 1. The Inertial Navigation System.
- 2. The Doppler Radar.
- 3. The Air Data Computer.
- 4. The Search Radar.



The inertial navigation system provides heading (azimuth), attitude (roll and elevation), and velocity increments in the X, Y, and Z directions. The doppler radar is a velocity sensor that utilizes the doppler shift principle to measure ground track speed and drift angle. The ground speed and drift angle derived from the doppler radar along with the true heading from the inertial navigation system are used to calculate the direction and magnitude of the wind. The air data computer uses the ambient static pressure and ram air pressure from the pitot tube to calculate corrected static pressure, pressure altitude, and mach number. Outputs from the air data computer are used to damp the raw velocities from the inertial system. The search radar provides target aximuth, range, and elevation signals to the navigation computer. The search radar elevation along with the aircraft elevation from the inertial system are used to measure the radar depression angle (look down angle to target from flight path vector). The depression angle and search radar slant range are used to compute ground distance to the target independent of altitude.

As the navigation computer calculates new incremental distances for each time increment, the present position is continuously updated. After each update, the executive computer is interrupted and the current value is passed to



it. If the navigation computer is functional, the aircrew will have current present position information independent of the status of the executive computer.

B. THE BALLISTICS COMPUTER

The ballistics computer is provided with the most current estimates of position and velocity of the aircraft. It is also provided with the weapon type selected by the aircrew. The ballistics algorithm computes the down range travel and time of fall for the weapon based on the airspeed, dive angle and altitude of the aircraft. The ballistics algorithm will be discussed in depth in Chapter III.

C. THE EXECUTIVE COMPUTER

The executive computer displays the information generated by both the ballistics and navigation computers. The executive computer is also responsible for issuing steering commands to the autopilot and firing pulses to the weapons release mechanism. The most important task the executive computer performs is to extrapolate a predicted weapon release point based on a time history of the position and velocity of the aircraft. Thus, even though the other two computers can operate independently, the aircrew cannot make a computer delivery without the executive computer.



D. THE COMMUNICATIONS SCHEME

Because of the inherent hierarchy between the computers involved, the master-slave type of multiprocessing is the most suitable and simple form to implement. The navigation and the ballistics computers act as peripheral devices of the executive computer, resulting in a one way interrupting scheme. The only computer which has to be interrupted is the executive computer. The navigation and ballistics computers are the dedicated slaves which asynchronously interrupt the executive computer.



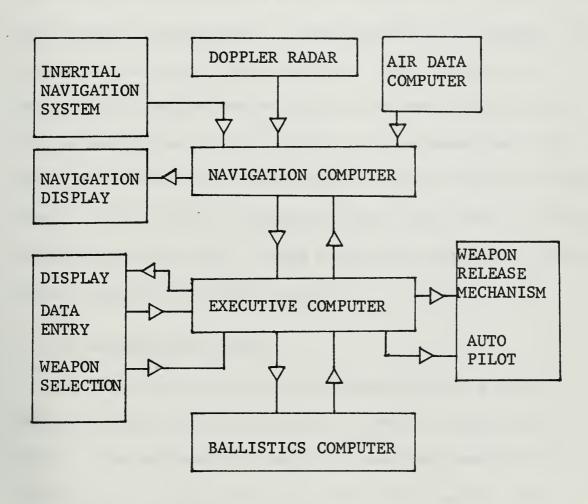


FIGURE 1. ORGANIZATION OF MULTIPLE PROCESSOR AIRBORNE TACTICAL WEAPONS SYSTEM



III. BALLISTICS PROBLEM

Since the earliest days of aerial warfare, the heart of the ballistics problem has been to drop a bomb from an airborne weapon platform and to consistently hit a target. The problem still exists today; however, with the aid of a computer the problem can be quickly and accurately solved. This necessitates the development of a mathematical model which approximates the actual path a weapon travels through space. Statistically, projectiles have been shown to follow predictable paths which behave very much like freely falling bodies described by Isaac Newton.

A. THE MATHEMATICAL MODEL

The ballistics problem can be described as a body falling through space according to Newton's second law of motion. The mathematical model or equation governing the trajectory of a ballistics projectile is a second order differential equation:

$$ma = mg - c |v|v$$
 (1)

where: m = mass of the body

a = acceleration vector

g = acceleration due to gravity



- c = drag coefficient
- v = velocity vector

The term c(|v|v) is the drag due to the air resistance of the body and is proportional to the square of the velocity.

At the time the projectile is released from the aircraft it will have an initial position and velocity. The differential equation together with the initial conditions uniquely determines the trajectory of the projectile.

This particular mathematical model was chosen for two reasons:

- 1. It approximates reality accurately.
- 2. The Navy publishes range and time of fall information in tables (NAVAIR 01-1C-1T-1) which uses this model.

The latter is the primary reason for using the ballistics tables as a standard or guideline in the verification of any new mathematical models or program implementations.

B. BASIC ASSUMPTIONS

Unfortunately there is no known analytical solution for this set of differential equations. However, a simpler problem does have an analytical solution. Thus, for the sake of simplicity and to aid in the discussion of this solution, let us first assume:



- 1. Level, non-accelerated flight.
- 2. Bombs are not ejected from the bomb rack.
- 3. No forward firing ordnance.
- 4. Drag/wind resistance is negligible.
- 5. Time of release is time = 0.

In addition, this discussion also assumed:

- 6. The Earth is flat and non-rotating.
- 7. The gravitational attraction, g, is constant.

C. DERIVATION OF DOWN RANGE TRAVEL

Rewriting equation (1) with respect to the time derivatives of the position vector, say u, the equation becomes:

$$m\ddot{u} = mg - c |u| u$$
 (2)

By letting u = (x,y), equation (2) is transformed into a system of two equations describing motion in a two dimensional coordinate system.

$$m\ddot{x} = \left[-c \sqrt{\dot{x}^2 + \dot{y}^2} \right] \dot{x} \tag{3}$$

$$m\ddot{y} = -mg - \left[c \sqrt{\dot{x}^2 + \dot{y}^2}\right] \dot{y}$$
 (4)

Initial conditions for the system are derived from the release position and velocity vectors of the aircraft:



Position Velocity
$$x(t_0) = r_x(t_0) \qquad \dot{x}(t_0) = \dot{r}_x(t_0)$$

$$y(t_0) = r_y(t_0) \qquad \dot{y}(t_0) = \dot{r}_y(t_0)$$

As a result of the assumption that air resistance is negligible and the aircraft's velocity and position determine the initial conditions, equations (3) and (4) are simplified to

initial conditions

$$m\ddot{x} = 0$$
 $\dot{x}(0) = \dot{r}_{x}(0)$ $x(0) = r_{x}(0)$
 $m\ddot{y} = -mg$ $\dot{y}(0) = \dot{r}_{y}(0)$ $y(0) = r_{y}(0)$

The solution of these two equations is obtained by dividing by the constant m, integrating twice with respect to time, and solving for the constants of integration.

$$\int \dot{x}(t) dt = \int 0 dt \implies \dot{x}(t) = c_1$$

$$\int \dot{x}(t) dt = \int c_1 dt \implies x(t) = c_1 t + c_2$$
when $t = 0$:
$$c_1 = \dot{x}(0) = \dot{r}_x(0)$$

$$c_2 = x(0) = r_x(0)$$

therefore,

$$x(t) = \dot{r}_{x}(0)t = r_{x}(0)$$
 (5)

where:



x(t) = total down range travel after release.

 $r_{x}(0)$ = initial displacement x(0) of weapon at the time of release, usually 0.

 $\dot{r}_{x}(0)t$ = down range travel due to initial velocity.

$$\int \dot{y}(t) dt = \int -g dt \qquad \Rightarrow \qquad \dot{y}(t) = -gt + c_1$$

$$\int \dot{y}(t) dt = \int -gt + c_1 dt \qquad \Rightarrow \qquad y(t) = -\frac{1}{2}gt^2 + c_1t + c_2$$
when $t = 0$:
$$c_1 = \dot{y}(0) = \dot{r}_y(0)$$

$$c_2 = y(0) = r_y(0)$$

therefore,

$$y(t) = -\frac{1}{2}gt^{2} + \dot{r}_{y}(0)t + r_{y}(0)$$
 (6)

where:

y(t) = the height above the ground at any time t.

 $r_y(0)$ = initial altitude y(0) of weapon at time of release.

 $\dot{r}_y(0)$ t = altitude loss/gain due to aircraft's initial vertical velocity.

 $-\frac{1}{2}gt^2$ = altitude lost due to gravity.

If the time from weapon release to weapon impact, or time of fall, is known, the ballistics problem is reduced to determining the down range travel, $x(t^*)$, given its initial-velocity, $\dot{r}_x(0)$ and the time of fall, t^* . Time of



fall can be found by setting equation (6) to 0 and solving for the positive root of t.

$$-\frac{1}{2}gt^2 + \dot{r}_y(0)t + r_y(0) = 0$$

Using the quadratic formula:

$$t* = \begin{bmatrix} -b & -\sqrt{b^2 - 4ac} \end{bmatrix} / 2a$$

where:

$$a = -g/2$$

$$b = \dot{r}_{y}(0)$$

$$c = r_v(0)$$

$$t* = \frac{\dot{r}_y(0) + \sqrt{(r_y(0))^2 + 2gr_y(0)}}{g}$$

By substituting time of fall, t*, into equation (5)

$$x(t*) = \dot{r}_{x}(0)t* + r_{x}(0)$$

down range travel, DRT, can be calculated.

DRT =
$$x(t^*) = \dot{r}_x(0) \frac{\dot{r}_y(0) + \sqrt{(\dot{r}_y(0))^2 + 2gr_y(0)}}{g} + r_x(0)$$

Since the coordinate system is arbitrarily placed, assume initial displacement, $r_{\rm x}(0)$, to be zero at the time of release. Also, since level non-accelerated delivery is



assumed, initial vertical velocity, $\dot{r}_y(0)$, is zero. The expression for down range travel now becomes:

DRT =
$$r_x(0) \sqrt{\frac{2r_y(0)}{g}}$$

In reality the problem must take into account the constraints placed on the problem earlier. Wind, drag, vertical velocity, and non-level delivery prarmeters make the solution more difficult. Solution in this case is accomplished numerically using a second order Runge-Kutta scheme on a digital computer.



IV. MICRO-COMPUTERS IN AIRBORNE TACTICAL WEAPON SYSTEMS

The question of whether or not micro-computers are feasible in an airborne tactical weapon systems environment was approached by first translating the ballistics algorithm into two high level languages, FORTRAN and PIM. (See Ref. 3). The FORTRAN version was executed on the IBM system 360 and the PIM version was executed on the INTEL 8080 micro-processor. Thus, if the same algorithm is executed on two different machines, the INTEL 8080, whose floating point mantissa has 16 bits and the IBM 360, whose floating point mantissa has 21-24 bits, then the differences in results can only be attributed to the difference in the precision of the two machines.

A. THE QUESTION OF ACCURACY

The solution of the ballistics problem requires solving a set of four differential equations numerically on a digital machine. This necessitates numerous arithmetic operations, including multiplication and division.

Since micro-computers presently lack hardware multiply and divide functions, a software package capable of performing floating point arithmetic operations must be used.

However, this requires a significant amount of additional



computing time in the solution of the ballistics problem.

As a result an alternate design in computer architecture was explored which utilizes three INTEL 8080 micro-processors instead of one, as in the mini-computer systems. In this multiple micro-processor system, each processor is dedicated to each of the primary functions of the system: executive, navigation, and ballistics computations.

The floating point mathematical package used in conjunction with the ballistics micro-processor uses a three byte binary representation with a 16 bit mantissa and an 8 bit exponent. The mantissa is left justified so that the most significant bit is always on and need not be stored, giving a full 16 bits of precision. The exponent is expressed as a power of 2 where the most significant bit serves as the sign bit. This three byte number is used instead of a conventional four byte scheme in order to reduce the time needed to perform the calculations necessary to solve the ballistics problem.

IBM's system 360 computer utilizes a four byte floating point hexadecimal number with a 21 to 24 bit mantissa and an 8 bit exponent. Since this method requires the first byte to have the value 1 to F (hexadecimal), the precision of the mantissa can vary between 21 bits (when the leading three bits are zero) and 24 bits. The exponent is expressed



as a power of 16 and also contains the sign bit. An obvious advantage the IBM 360 has is that it does floating point arithmetic in hardware which makes it approximately two orders of magnitude faster than the software version.

The INTEL 8080 micro-computer uses 8 bit operations and has the option of using double precision (16 bit) operations. The double precision operation permits multiplications and division to be performed as sequences of 16 bit additions and subtractions. Then, if the mantissa is kept left justified (16 bit precision), the double precision feature can be used to maintain 16 bits of precision throughout the calculation. If more precision is desired, such as 21 or 24 bits of precision, a quantum jump in execution time can be expected because of the additional computer cycles required.

It is the intention of this thesis to show that the loss of one byte of precision will not significantly affect the results of the ballistics solution. The accuracy with which a weapon is delivered depends greatly upon the accuracy and precision of the sensor supplied information. The loss of one byte of precision only affects the sixth most significant digit which is an order of magnitude more precise than most of the input sensors on board attack aircraft. This then is the motivation to compare the



results of a FORTRAN ballistics algorithm executed on the IBM system 360 with the same algorithm translated into PLM and executed on the INTEL 8080 micro-computer.

B. THE QUESTION OF SPEED

The second area of interest is the question of speed. At the time this thesis was written, an LSI (large scale integration) "chip" existed which could perform the floating point multiply and divide operations at the cost of \$270. However, due to budgetary constraints this equipment was not readily available for experimentation. The hardware floating point "chip" can execute approximately 100 times faster than the software floating point package. For example, a multiply operation in the software package takes approximately 600 microseconds to execute whereas the hardware package executes a multiplication in 6 microseconds. hardware multiply and divide operator was also developed and constructed as a micro-computer course project at the Naval Postgraduate School and was demonstrated to function at 60 microseconds.

By interrupting the program during execution and recording the location of the program address register, it was determined that the ballistics program spends about 92% of its execution time in the floating point package. According



to Jupin (Ref. 3) the execution time of each solution was proportional to the calculated time of fall. These results were confirmed by executing nearly 1800 separate calculations. The time to calculate the predicted release point proved to be about 10% of the calculated time of fall. Linhares (Ref. 2) was able to show that the ballistics algorithm was fast enough for certain initial conditions, however for high airspeeds and low altitude release conditions his extrapolation technique was not usable.

A substantial amount of the time the program spends in the floating point package (about 92%) is spent in the multiply and divide procedures. Using either the commercially produced "chip" or the locally constructed hardware multiply and divide operator, a significant reduction in execution time would result. Although this thesis will not answer the question of speed with an unqualified yes, it supports the finding that the ballistics processor is fast enough.



TRIAL	TOTAL NUMBER OF INTERRUPTS ATTEMPTED	NUMBER OF INTERRUPTS PROGRAM WAS IN FLOATING POINT PACKAGE	PER CENT OF TIME PROGRAM SPENT IN FLOATING POINT PACKAGE
1	323	289	89.5
2	343	319	92.0
3	367	346	94.3
TOTAL	1033	954	92.4

TABLE 1. AMOUNT OF TIME BALLISTICS PROGRAM SPENDS IN THE FLOATING POINT PACKAGE



V. VERIFICATION OF DOWN RANGE TRAVEL

Two separate verifications of down range travel were made, using the ballistics tables as a "standard." First, the FORTRAN and PLM versions of the mathematical model, previously discussed, were tested against the ballistics tables. Second, observed data was compared against the FORTRAN version for accuracy in time of fall and down range travel. The FORTRAN version was executed on the IBM 360 (32 bit machine), while the PLM version was executed on the INTEL 8080 micro-computer utilizing a 24 bit floating point mathematical package on an 8 bit machine.

A. THE BALLISTICS ALGORITHM - FORTRAN VS. PLM

A straight forward comparison between the FORTRAN version and the PLM version of the ballistics algorithm was made contrasting the down range travel and time of fall. An input/output interface was written to the PLM program so that data could be read from a floppy disk and the results written onto the same device. The floating point package was also modified in order to execute on the INTEL 8080 and a logic error in the multiply procedure was corrected. The FORTRAN program (Ref. 3) was virtually unchanged, however a statement was added to the TRAJ subroutine to



patch a logic error affecting the second stage trajectory calculation.

B. OBSERVED DATA VS. BALLISTICS TABLES

Since the ballistics tables are considered a "standard" against which various types of ballistics results are compared for validity, a comparison between observed results, obtained from the A6-E experimental data, and the ballistics tables (NAVAIR 01-1C-1T-1) was desired to establish a correlation between the two. However, several problems were encountered in making the comparison.

1. Predicted Down Range Travel

The initial conditions of the observed data are not compatible with those of the ballistics tables. The observed data has initial conditions composed of various dive angles, altitudes, and airspeeds, whereas the ballistics tables' initial conditions are multiples of 50 and 100 for altitude and airspeed, and zero for dive angle (only considering level delivery). Two apparent solutions to this difficulty are (1) the error sensitivity tables and (2) interpolation of the ballistics tables. However, both have disadvantages.

The error sensitivity tables failed to help because the corrections are based on maintaining a constant sight



line (mil setting) rather than keeping down range travel constant. For example, in a level delivery situation the error sensitivities table assumes a constant sight line and varies the down range travel by affecting corrections to the altitude. Therefore, if the altitude is higher than planned and the sight line is maintained, the hit will be short. However, the problem using observed data requires a constant down range travel to target, applying correction for altitude and airspeed. Therefore, the error sensitivity tables could not effectively be used in this case.



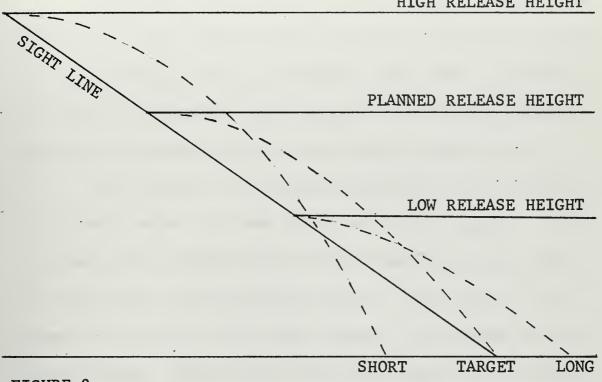


FIGURE 2. EFFECTS OF BALLISTICS TABLES ERROR SENSITIVITIES FOR HEIGHT

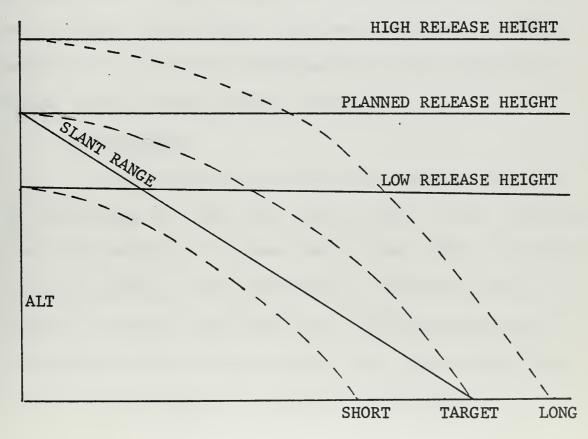


FIGURE 3. DESIRED ERROR SENSITIVITIES FOR BALLISTICS PROBLEM



The ballistics tables are not linear. Second and even third order interpolation would be required to determine down range travel on anything other than a cardinal altitude or airspeed. This process is extremely tedious and time consuming and was abandoned as impractical.

As a result, an accurate substitute for the ballistics table value for down range travel was sought. Using the ballistics tables for the MK-76 mod-5 practice bomb, 96 data points were selected from all dive angles, airspeeds, altitudes, and tested against the FORTRAN ballistics program. Time of fall and down range travel compared favorably with a mean error of 0.23% for down range travel and 0.11% for time of fall. This was considered accurate enough to be used as an approximation for the ballistics tables' value for down range travel.

2. Wind Effect

The wind plays an important and rather subtle role in determining the total down range travel. First, assuming a no wind condition, the down range travel, DRT, is computed in the direction of the true heading. Since no wind is present to deflect the projectile, the true heading and ground track will coincide and the down range travel along the true heading and ground track will be equal.



Now consider the wind. If a coordinate system (x,y,z) is introduced such that the air does not move with respect to the coordinate system, then an aircraft in this air mass has velocity with respect to the air mass and its heading will be the true heading. Because the air mass moves with respect to the ground with the wind velocity, $w = (w_x, w_y, w_z)$, the Earth fixed coordinate system (x',y',z') is related to (x,y,z) by the following equations:

$$x' = x + w_x t$$

$$y' = y + w_y t$$

$$z' = z + w_z t$$

Considering only the horizontal wind, the vertical wind, w_z , becomes zero. Down range travel can now be determined given an initial altitude, z, and air velocity, V, and will be in the direction of the true heading. To determine the point (x',y') in the Earth fixed coordinate system at which the weapon lands requires:

$$x' = x(tof) + w_x * (tof)$$

 $y' = y(tof) + w_y * (tof)$

For example, given the same initial conditions, down range travel will be computed, as before, in the direction of the true heading. However the aircraft will drift with the



wind and will actually move across the ground on a different heading, called ground track. Since DRT is computed along true heading, the projection of DRT onto the ground track is that distance the projectile will travel due to the initial conditions alone.

PROJECTED DRT = cos(TH-GT) * DRT

The down range component (along the ground track) of DRT due to wind is computed as follows:

X-COMP = -cos(WDIR-TH) * WKTS * TOF 1.6867 where:

X-COMP = down range component of DRT due to wind

WDIR = true wind direction

TH = true heading

WKTS = wind speed in knots

TOF = time of fall in seconds

1.6867 = conversion from knots to ft/sec

This distance added to the PROJECTED DRT gives the WIND

CORRECTED DRT which is the total DRT the projectile will

travel in a moving air mass.

WIND CORRECTED DRT = X-COMP + PROJECTED DRT



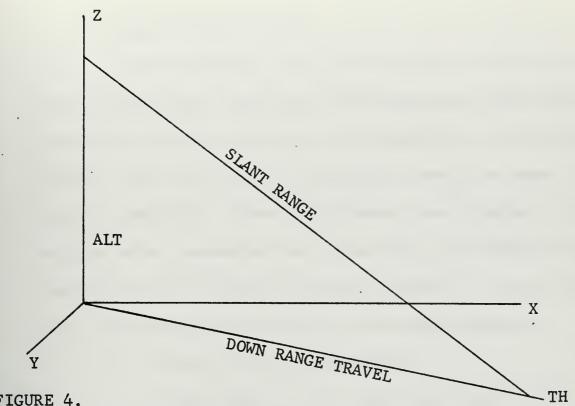


FIGURE 4.
NO WIND SOLUTION TO DOWN RANGE TRAVEL

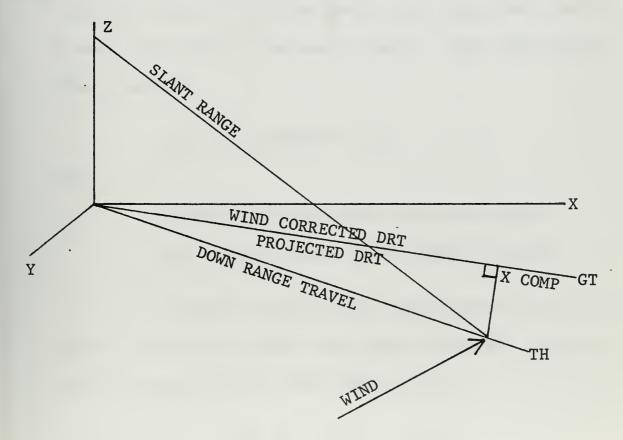


FIGURE 5. WIND CORRECTED SOLUTION TO DOWN RANGE TRAVEL



3. Observed Slant Range

The experimental data taken from the A-6E aircraft during actual drop conditions provides many useful delivery parameters, sensor readings, and intermediate calculations. However, the one piece of information needed to make the desired comparison, down range travel, was missing. But various other parameters were available, and DRT could be reconstructed by several different methods. The problem was to determine which method is the most accurate.

a. Method for DRT Calculation

Slant range to the target and search radar depression angle (look down angle from flight path vector) is made available by the search radar. Down range travel simply becomes

DRT = cos(DEPANG) * SR

where:

DEPANG = search radar depression angle
SR = search radar slant range to target

A second method takes advantage of the aircraft's present position altitude and target altitude to compute vertical separation and down range travel.

VERT SEP = PPA - TGT ALT

DRT = cos arsin(VERT SEP / SR) * SR



where:

VERT SEP = vertical separation

PPA = inertial derived present position altitude

TGT ALT = target altitude

A third and final method of computing down range travel uses the vertical separation generated by the ballistics program on board the A-6E aircraft.

DRT = cos arcsin(VERT SEP / SR) * SR

b. Error Analysis

An error analysis was conducted to determine which method would produce the most accurate value for DRT. This would yield a maximum error bound which can be expected in DRT due to this method of calculation. The third relationship proved to have the smallest error bound and was eventually used to reconstruct the down range travel from the experimental data.

For example, the maximum error bound on the relationship

DRT = cos arsin(VERT SEP / SR) * SR

is the sum of the partial derivatives of DRT multiplied by their tolerances.



$$\left| \triangle DRT \right| = \left| \frac{\partial DRT}{\partial VERT SEP} \right| \left| \triangle VERT SEP \right| + \left| \frac{\partial DRT}{\partial SR} \right| \left| \triangle SR \right|$$

The third method of computing down range travel will serve as an example for this procedure.

The partial derivative of DRT with respect to VERT SEP becomes:

$$\frac{\partial DRT}{\partial VERT SEP} = \frac{SR}{\sqrt{(SR)^2 - (ALT)^2}}$$

The partial derivative of DRT with respect to ALT becomes:

$$\frac{\partial DRT}{\partial ALT} = \frac{-ALT}{\sqrt{(SR)^2 - (ALT)^2}}$$

the error bounds are:

VERT SEP = ± 0.5 feet (rounded to the nearest foot)

 $SR = \pm 5.0$ feet (rounded to the nearest 10 feet)

Therefore, the maximum error bound that can be expected from rounding error of actual delivery data becomes:

$$\Delta DRT = \frac{0.5(SR) + 5.0(ALT)}{\sqrt{(SR)^2 - (ALT)^2}}$$

This analysis was performed using all three methods of constructing down range travel. The third method was found to be the most accurate with a maximum



error bound of 5.2 feet and was the method used to construct the down range travel from the freeze data.

4. Hit Distances

The hit coordinates of the experimental data were not utilized because the computer bases its calculations on the location of the search radar cursors. If the cursors are not properly placed, the weapon, most likely, will not hit the target. However, for the slant range measured by the search radar set, the down range travel and time of fall will be calculated accurately and the weapon will hit the ground in the proximity of the cursor placement.



VI. PRESENTATION OF RESULTS

A. FORTRAN VS. PLM

Appendix A contains the results of comparing identical ballistics algorithms: a FORTRAN program which is a Naval Postgraduate School modification of the Naval Weapons Center BOEING algorithm, and a PLM version of the same algorithm (Ref. 3). The difference between the procedures is that the FORTRAN program uses the standard IBM 21 to 24 bit mantissa for its floating point number, whereas the PLM version uses a 16 bit mantissa.

A summary of the results is presented in Table 2.

With the exception of weapon number five (MK-76 MOD-5)

the largest average difference in down range travel (DRT)

was 1.6 feet and the maximum absolute difference in DRT

was 17 feet, which occurs when DRT is 8,332 feet. Weapon

number five is suspected to have a coefficient error,

although none have been discovered.

Overall these results indicate that the sixteen bit mantissa is sufficiently accurate to perform the ballistics algorithm.



WEAPON ID NUMBER	AVERAGE PER CENT IN DRT (feet)	AVERAGE PER CENT IN TOF (seconds)	MAXIMUM PER CENT IN DRT (feet)	MAXIMUM PER CENT IN TOF (seconds)
4	0.0191	0.1733	0.18	0.68
5	0.1150	0.5622	0.72	2.24
6	0.0070	0.0745	0.03	0.31
7	0.0200	0.0200	0.17	0.60
8	0.0152	0.1318	0.13	0.66
9	0.0160	0.0886	0.03	0.58
10	0.0141	0.0830	0.13	0.56
11	0.0158	0.1066	0.12	0.55
12	0.0128	0.0744	0.16	0.41
13	0.0077	0.0498	0.05	0.28
14	0.1661	0.0938	0.16	0.59
15	0.0102	0.0728	0.06	0.37
16	0.0151	0.0899	0.20	0.62
17	0.0062	0.0269	0.02	0.13
18	0.0103	0.0056	0.03	0.02
20	0.0072	0.0167	0.02	0.04
21	0.0147	0.0105	0.04	0.02
22	0.0107	0.0080	0.03	0.02

TABLE 2. SUMMARY OF FORTRAN VS. PLM RESULTS



B. BALLISTICS TABLES VS. FORTRAN RESULTS

The ballistics tables were compared against the results of the FORTRAN version of the ballistics algorithm. Appendix B contains the results for a variety of initial conditions and weapon types. The results indicate substantial discrepancies between the ballistics tables and the FORTRAN program. The latest version of the Naval Weapons Center's ballistics algorithm (including revised drag coefficients) still does not resolve these differences. Table 3 gives a summary of the data in Appendix B. Weapon number five (MK-76 MOD-5), which was used in the experimental data shows reasonable accuracy for the range of parameter values used in the experiment.



WEAPON ID NUMBER	AVERAGE PER CENT IN DRT (feet)	AVERAGE PER CENT IN TOP (seconds)	MAXIMUM PER CENT IN DRT (feet)	MAXIMUM PER CENT IN TOF (seconds)
4	0.1751	0.1758	0.67	0.49
5	0.2260	0.1135	0.93	0.51
6 *	19.734	20.092	41.14	38.30
7	0.5498	0.8586	2.02	1.78
8 *	0.0986	0.1117	0.42	0.34
9	0.1396	0.1743	0.72	1.02
10	0.4749	0.7935	2.03	2.89
11	0.0777	0.1118	0.50	0.44
12	0.1102	0.3033	0.48	0.86
13	0.0526	0.0892	0.13	0.76
14	0.1050	0.1937	0.32	0.67
15	0.1075	0.0456	0.26	0.18
16	0.3688	0.2456	1.58	1.15
17 *	0.0521	0.0977	0.20	0.62
18	3.3773	1.6483	27.34	9.20
20	1.1159	1.5200	7.46	10.32
21	2.8479	4.9681	8.62	23.81
22	3.5271	1.5409	28.23	8.40

^{*} used invalid drag and mach coefficients

TABLE 3. SUMMARY OF BALLISTICS TABLES VS. FORTRAN RESULTS



C. EXPERIMENTAL DATA VS. FORTRAN ALGORITHM

Experimental data gathered by bombardiers from Naval Air Station Whidbey Island at Boardman bombing range is given in Appendix C. Because the ballistics table and the FORTRAN algorithm agree reasonably well, the FORTRAN algorithm was used in place of the ballistics tables for convenience. The down range travel and time of fall were calculated from:

- 1. The experimental data.
- 2. The FORTRAN algorithm with the old drag coefficients.
- 3. The FORTRAN algorithm with the new drag coefficients. The calculations are described in section (5.B.2, Wind Effect), and the results are summarized in Table 4.

The substantial discrepancy between the experimental data and the results of the ballistics algorithm cannot be dismissed. Because the ballistics algorithm's down range travel agrees with the official Navy ballistics tables to within 0.2%, the 12% discrepancy in DRT leads to the conclusion that either the instrumentation on many different aircraft indicated erroneous readings, or the behavior of the weapons is substantially different from the behavior described by the ballistics tables. Additional data under more precise initial conditions would have to be gathered before any definite conclusions can be drawn.



	OLD COEFFICIENT	NEW COEFFICIENTS
AVERAGE ABSOLUTE DIFFERENCE IN DOWN RANGE TRAVEL	247.00	247.00
AVERAGE ABSOLUTE DIFFERENCE IN TIME OF FALL	0.1299	0.1573
AVERAGE PER CENT DIFFERENCE IN DOWN RANGE TRAVEL	11.973	12.435
AVERAGE PER CENT DIFFERENCE IN TIME OF FALL	1.659	2.091

TABLE 4. SUMMARY OF BALLISTICS TABLES VS. ACTUAL DELIVERY DATA



VII. CONCLUSION

The radical cost reductions in computer hardware brought about by large scale integration (LSI) have introduced an opportunity to construct micro-computer based airborne tactical systems which reduce the hardware costs by at least an order of magnitude. To establish the feasibility of constructing such a system requires that two questions be answered in the affirmative.

- 1. Is the computation sufficiently accurate?
- 2. Is the computation fast enough to satisfy real time requirements?

This study concentrated on the first question. As the results indicate, the 16 bit floating point mantissa is sufficiently accurate for the ballistic calculations.

As a byproduct, the BOEING-Naval Weapons Center algorithm was compared with the published ballistics tables.

Although some of the weapons displayed close agreement, others revealed substantial discrepancies which remain unresolved.

The most significant and unexpected finding is related to the experimental data generated by the bombardiers based at Naval Air Station Whidbey Island. If the initial



conditions recorded by the aircraft's instruments are used to predict where the weapon would impact the ground, then the ballistics tables predict that the weapons land consistently more than 10% short of where they actually landed. Either the recorded initial conditions are incorrect or the ballistics tables do not predict reality for this weapon.

Although the micro-computer is substantially slower in executing arithmetically complex tasks when compared to a mini-computer, several micro-computers can be used as dedicated machines for specific tasks. Such a distributed system can operate sufficiently fast to solve the real time problem.



APPENDIX A

This appendix compares the results of the FORTRAN and PLM versions of the ballistics algorithm. The absolute difference in down range travel and time of fall is presented.



00	T ERROR DIST	000000000000000000000000000000000000000
DS SL	PER CEN	000000000000000000000000000000000000000
00		
= ZN	RENCES DIST	1 1111 11111111111111111111111111111111
ΣZ ΣZ	DIFFE	
	N X	
0000	N VERSIC MODIFIED ALGORITH DIST	2 2 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2 2 2 2
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	FORTRAP NPS N BOEING TIME	1 1 1 1
40		
4 0.0027540 0.0 3.00	ERSION DIFIED LGORITHM DIST	1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
DKG1 = DKG2 = IREF = DMAX =	PLM NPS MO OEING A	1 111111111
FOR	ω	
EFFICIENTS 0.0039235 0.0	ALT	11 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
PON COEFF RM1 = 0.0 RM2 = 0.0	TAS	uuuuu4444nnnnuuuuu4444nnnnuuuuu44 oonnoonnoonnoonnoonnoonnoo oonnoonn
CFORN CFORN ITYPE	DEG	000000000000000000000000000000000000000



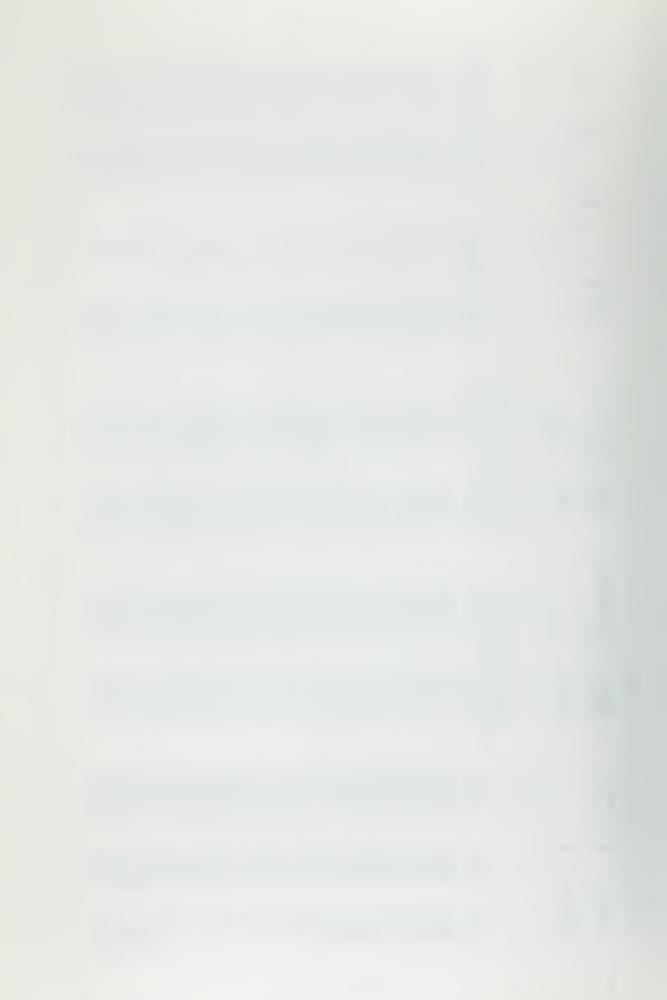
T ERROR DIST	000000000000000000000000000000000000000	•
PER CENTINE	00000000000000000000000000000000000000	•
ER ENCES DIST	9-MONO0-000NN	•
DIFFE		•
N E	·	
VERSI ODIFIED ALGORIT DIST	1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	7
NPS MOEING	のもつらららももももものもっちっとしてころっとっちょうできょうできらららららららららららららららららららららららららららららららら	•
8 T		
¥.		•
ERSION DIFIED LGORITH DIST	1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	1
A A A A A A	りらてのろらんももらよらてきよこしもとくりこうりこすのとまろうちょうてりのこうらうともあるようできらんようのほううりこうこうまとてってらしこうなってっている。)
PLW BOEINS	2 2 2 2 2 2 2 2 2 2 2 1 1 2 2 2 2 1 1 2	4
ALT		
S		•
TAS	44NNNNWWWWW4444NNNNWWWW4444NNNNWWWWWA444ANN NNOONNOONNOONNOONNOONNOONNOONNOO)
DEG		•



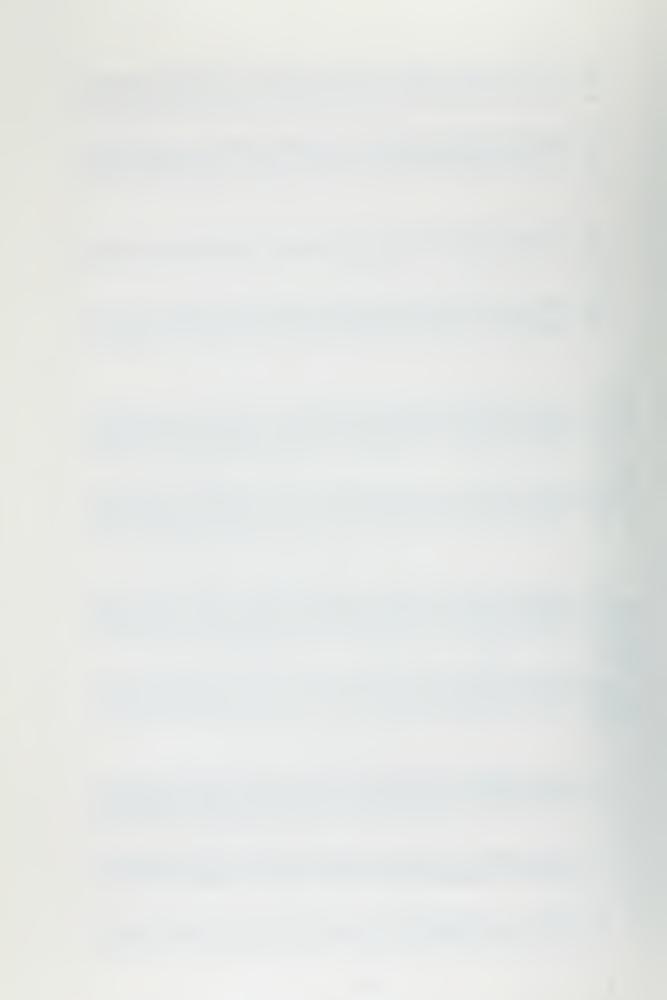
IT ERROR DIST	
PER CEN	
RENCES DIST	-000000-0000000000000000000000000000000
DIFFE	
MODIFIED ALGORITHM DIST	1 1117070808070707070707070707070707070707
FORTRA NPS BOEING TIME	1
DIFIED LGORITHM DIST	114 114 1010 1010 1010 1010 1010 1010 1
PLM NPS MOLE	2
ALT	4 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
TAS	
DEG	1 1 1 1 1 1 1 1 1 1



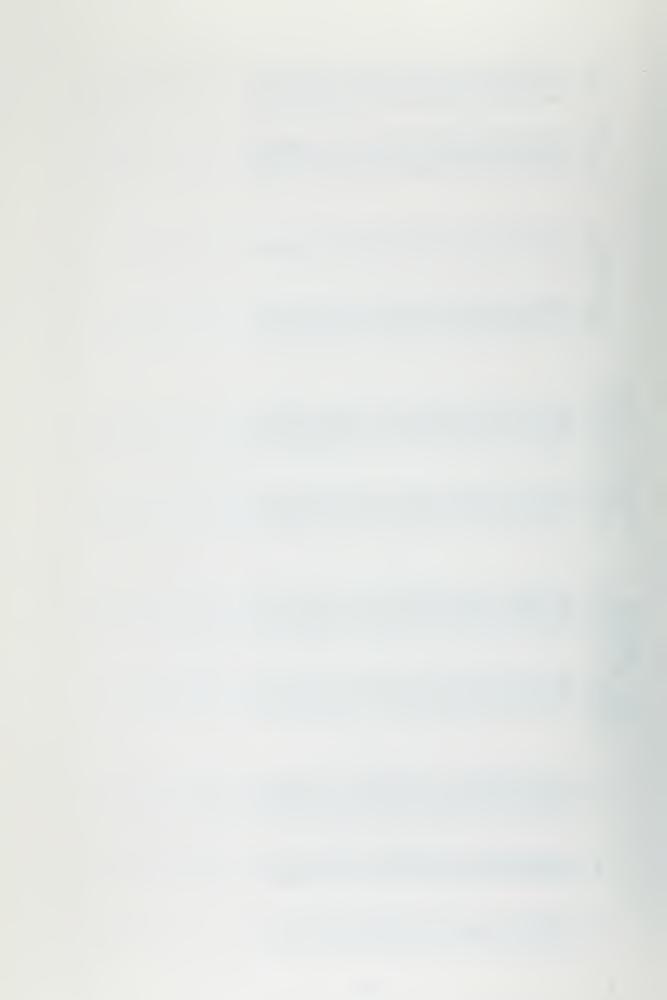
0.0 = 0.0 0.0 = 0.0	PER CENT ERROR TIME DIST	00000000000000000000000000000000000000
VMUZ =	DIFFERENCES TIME DIST	0000 0000 0000 0000 0000 0000 0000 0000 0000
	TRAN V PS MODI ING ALG	17-18 17-18 17-18 17-18 18-23 18-23 18-23 18-24 18-26 18
FOR IDNO 5 DKG1 = 0.0063648 DKG2 = 0.0	MAX = PAX = ING ACD	17.18 17.12 17.12 17.12 10.19 10.19 10.19 10.19 11.00.19
ICIENTS 039077	ALT	15 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4
WEAPON COEFF CFCRM1 = 0.0 CFORM2 = 0.0	BOTH = EG TAS	



ERROR	
PER CENT	00000000000000000000000000000000000000
RENCES	1 1111 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
DIFFE	00000000000000000000000000000000000000
MODIFIED ALGORITHM DIST	1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
FORTRAI NPS I BOEING TIME	2
ERSION DIFIED LGURITHM DIST	1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
PLM V NPS MO BOEING A	2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
ALT	1
TAS	
DEG	00000000000000000000000000000000000000



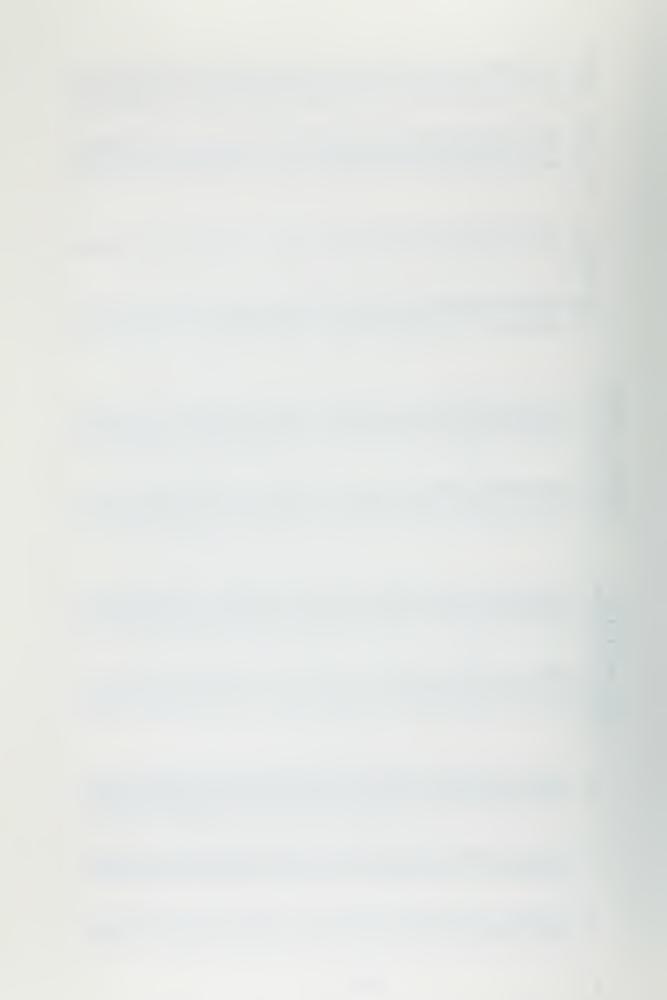
PER CENT ERROR TIME DIST	00000000000000000000000000000000000000
RENCES DIST	
OIFFE TIME	00000000000000000000000000000000000000
MODIFIED ALGORITHM DIST	1 04 100 100 100 100 100 100 100 100 100
FORTRAI NPS BOEING TIME	2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
ERSION DIFIED LGORITHM DIST	1040 1000 1000 1000 1000 1000 1000 1000
PLM NPS MO BOEING A	2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
ALT	42000000000000000000000000000000000000
TAS	nnumum4444nnnnumum4444nnnn nnoonnoonnoonnoonnoonn 0000000000
DEG	1 1 1 1 1 1 1 1 1 1



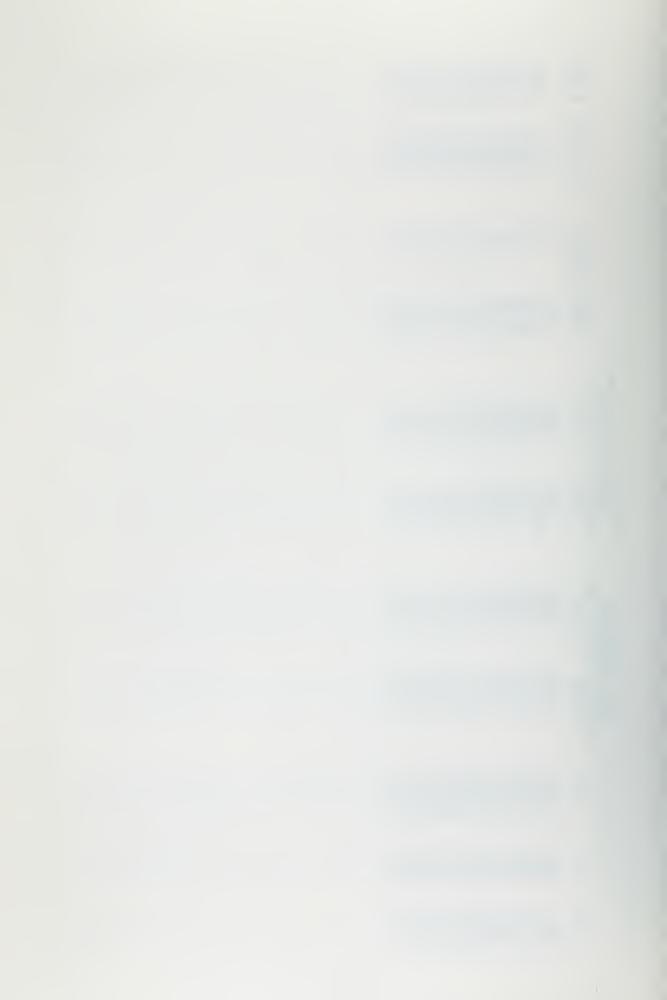
	000		T ERROR DIST	000000000000000000000000000000000000000
	DS SL		PER CEN	000000000000000000000000000000000000000
	00			
	= 71		RENCES	
	N N N N		DIFFE	000000000000000000000000000000000000000
			N F	
	00	0.00	N VERSIC MODIFIED ALGORITH	00000000000000000000000000000000000000
	DM1 =	VE = DTI =	FORTRAN NPS N BOEING TIME	1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
	99			
9	0.021266	2.00	ERSION DIFIED LGORITHM DIST	100886542000000000000000000000000000000000000
OR IDNO	DKG1 = DKG2 =	IREF DMAX =	PLM VI NPS MOI BOEING A	1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
S F				• • • • • • • • • • • • • • • • • • • •
FICIEN	00		ALT	1
PON COEF	$\frac{1}{2} = \frac{0}{0}$		TAS	www.444\n\n\n\w\w\w\\\\\\\\\\\\\\\\\\\\\
WEAPC	CFORM	I TYPE I BOTH	DEG	



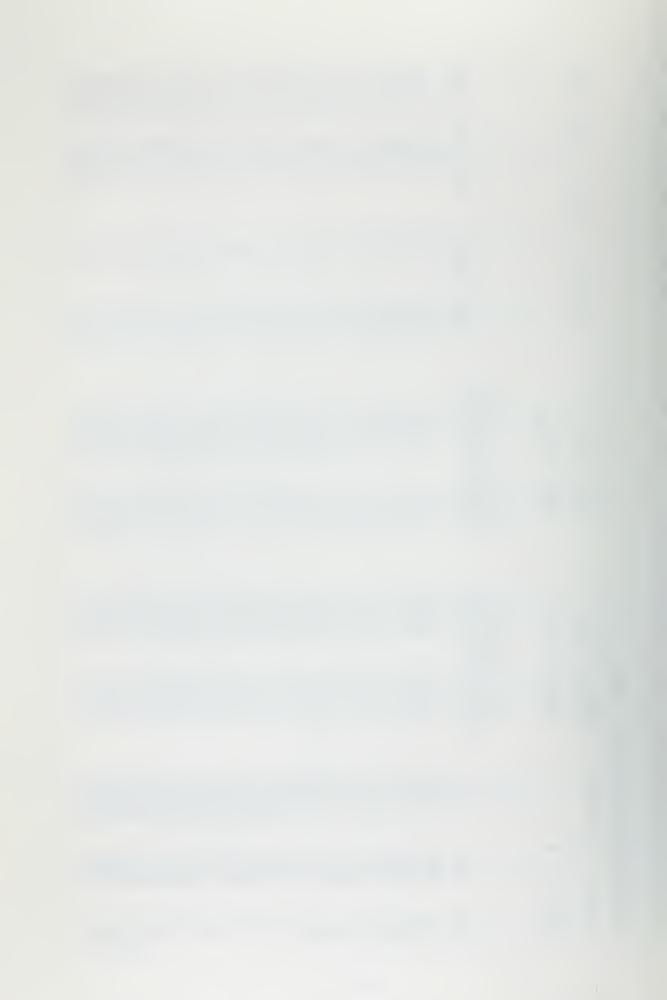
ERROR DIST	
PER CENT	00000000000000000000000000000000000000
RENCES	0-000-00000000000000000000000000000000
DIFFE	
ND I ON I H	
VERS ODIFIE ALGORI DIS	$\begin{array}{c} WLW \\ WLW$
FORTRAN NPS M BOEING TIME	1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
I FIED GORITHM DIST	### ### ### ### ### ### ### ### ### ##
PLM VER NPS MODI BOEING ALG	
ALT	######################################
TAS	
DEG	



T ERROR	?	•		•	•	ó	• •	•	•	00.0-	•	•	•	•
P ER CEN	Σ	-	0.00	.0	0	0	-	•	7	0	~	0	_	~
SI SI SI SI SI SI SI SI SI SI SI SI SI S	?			•	o	0	0-	•0-	0-	0	•	•	•	•
DIFFER	-	•		•	•	•	•	•	•	•	•	•	•	•
VERSION ODIFIED ALGORITHM	?	83	1785	55	ж 8	8	35	36	63	69	12	16	41	23
FORTRAN NPS M	Ξ	5.7	72.07	3.2	2.0	2.0	8° ×	9.8	6°2	8.6	3. 6	ထ္	8.7	φ.
RSION IFIED GORITHM	—	183	1785	55	88	98	35	36	63	69	12	26	41	23
PLM VE NPS MOD BOEING AL	Σ	9.9	20.32	3.2	7.0	1.9	7.8	0.8	6.2	8	8.6	8	8.6	6.
:	AL I	4500.	15000	15000	4000	15000.	5000	000	S	15000.	6500	000	-	000
	TAS	S	200 000	\circ	S	S	0	0	S	3	0	0	5	S
	DEG	4.	147	S O	9	9	9	9	9	9	9	9	9	9



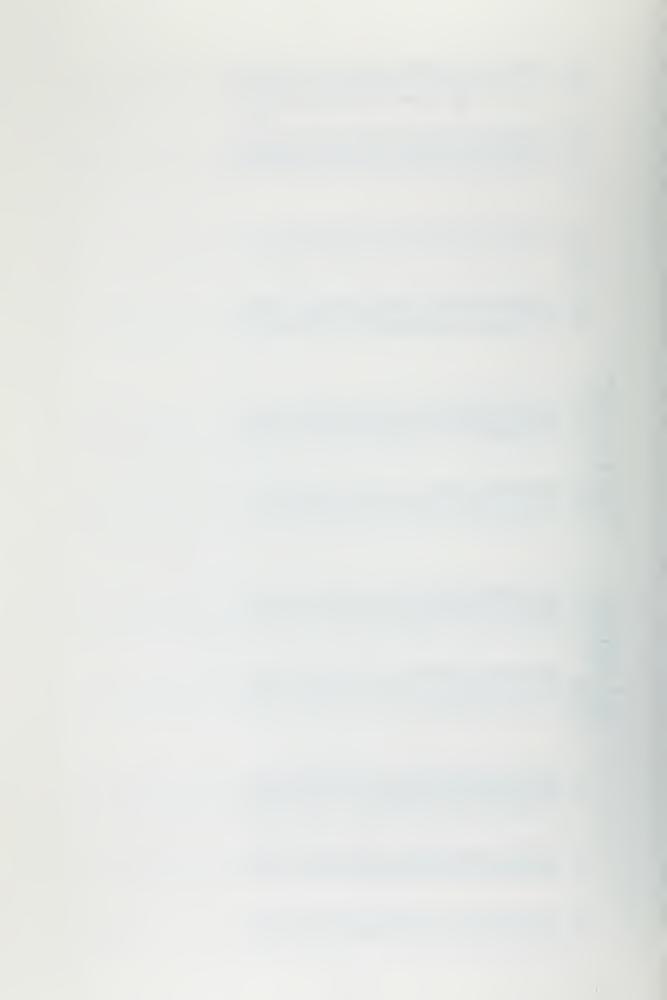
	DS = 0.0 SL = 0.0		ER CENT ERROR	00000000000000000000000000000000000000
	VMUZ = 0.		DIFFERENCES PE TIME DIST	00000000000000000000000000000000000000
	DM1 = 0.0 DM2 = 0.0	VE = 0.0 DTI = 3.00	FORTRAN VERSION NPS MODIFIED BOEING ALGORITHM TIME DIST	168.78 168.78 179.58 179.58 170.27 18.59 18.50 18.5
FOR IDNO 7	DKG1 = 0.0 DKG2 = 0.0	IREF = 1 DMAX = 5.00	PLM VERSION NPS MODIFIED BOEING ALGORITHM TIME DIST	16.77 16.77 170.258 170.256 180.992 180.992 180.992 180.992 180.992 180.992 180.992 180.992 180.992 180.992 180.992 180.992 180.992 180.992 180.992 180.992 180.992 180.992 180.993
FICIENTS	5703993		ALT	
WEAPON COEF	CFORM1 = 2. CFORM2 = 0.	ITYPE = -1 IBOTH = 1	DEG TAS	00000000000000000000000000000000000000



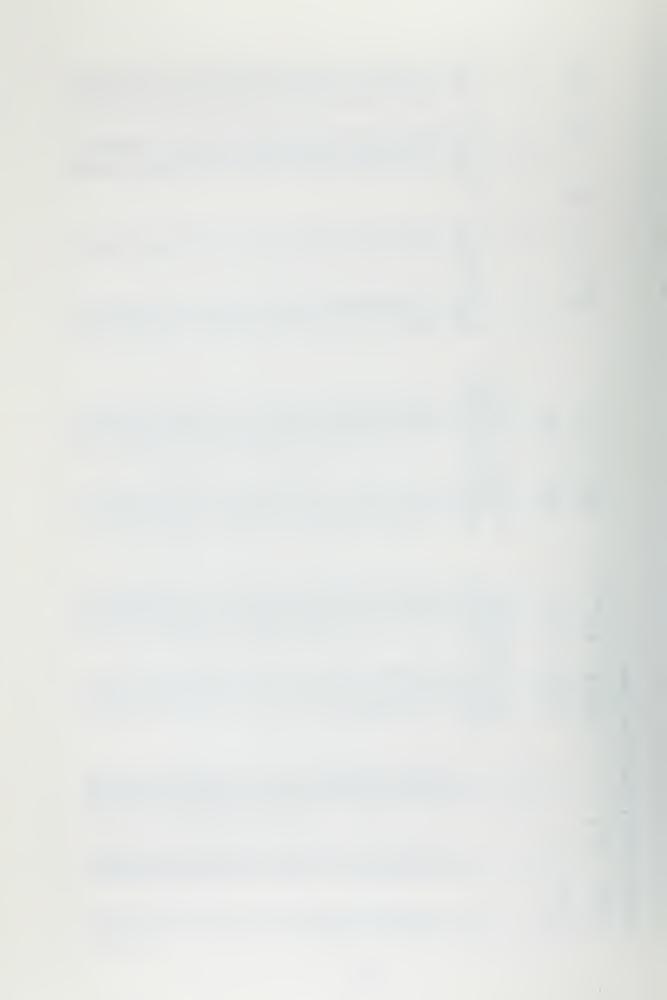
PER CENT ERROR TIME DIST	
RENCES	000004000000000000000000000000000000000
DIFFE	000000000000000000000000000000000000000
MODIFIED ALGORITHM DIST	1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
FORTRAI NPS BOEING TIME	1
ERSION DIFIED LGORITHM DIST	1 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
PLM V NPS MO BOEING A	1 2 2 1 1 2 2 2 2 1 1 2 2 2 2 2 2 2 1 1 2 2 2 2 2 2 2 1 1 2
_	
TAS	44NNNNWWWWW4444NNNNWWWWW4444NN NWOONNOONNCONNOONNOONNOONNOONNOO 0000000000
DEG	1

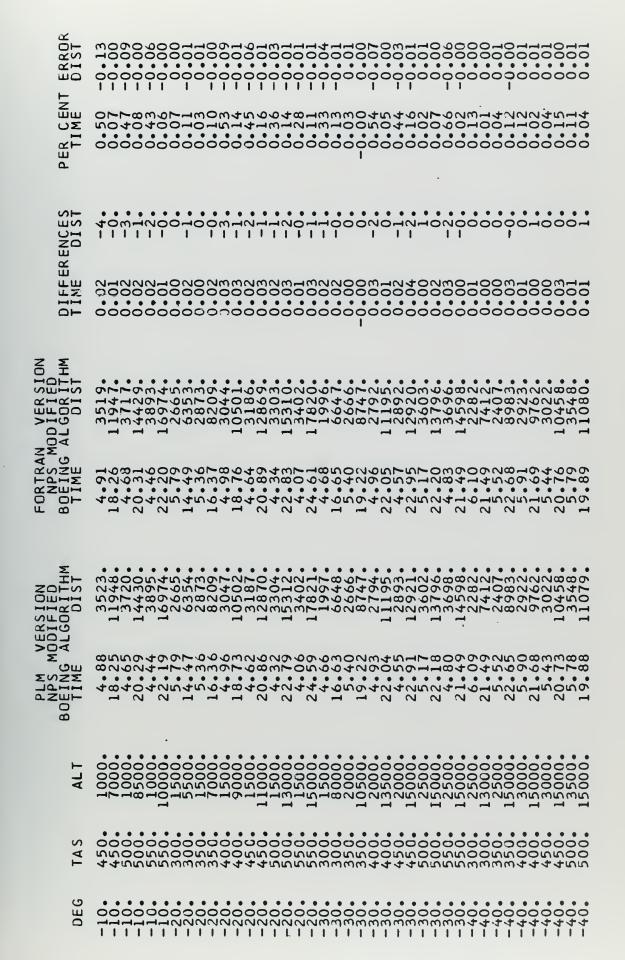


T ERROR DIST	000000000000000000000000000000000000000
PER CENT	00000000000000000000000000000000000000
RENCES	
DIFFER	######################################
ΖΣ	
VERSION MODIFIED ALGORITHM DIST	1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
FORTRAN NPS MI BOEING	1
RS ION IFIED GORITHM DIST	1 1 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
VE MOD AL	1733991920907937十5348985 5
PLM NPS BOEING	日 1 2 1 日 日 1 2 日 日 日 1 2 2 2 2 2 2 2 2
	•
ALT	4 11 2 1 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2
TAS	nnumum4444nnnnumum4444nnnn nnoonnoonnoonnoonnoonn 0000000000
DEG	



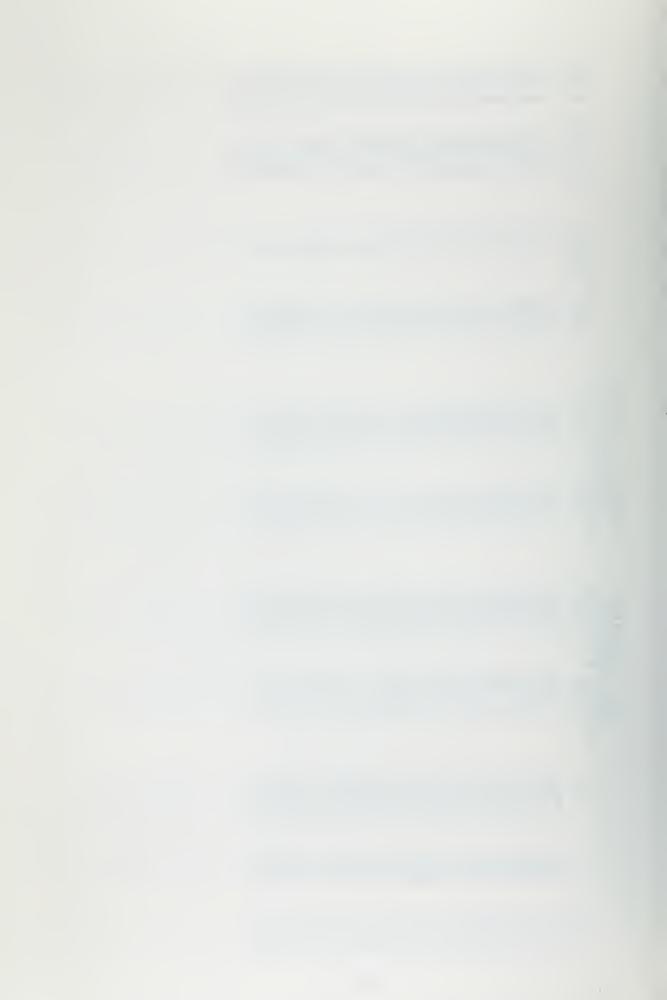
00	T ERROR DIST	000000000000000000000000000000000000000
DS SL	PER CENT	
90	∨⊢	
# 7 =	RENCE	
ΣZ ΣZ	DIFFE	
	N X	
00 00	VERSI ODIFIED ALGORIT DIST	1 1 1 1 1 1 1 2 2 2 2 0 0 0 0 0 0 0 0 0
DM1 = VE = 1	TRA PS ME	1 1 3 3 3 3 3 1 1111111111111111111111
0		
8 0.0097670 0.0	STO O	1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
KF 21 NO	Σ	8144641060004046802066910549599 9144641060104024001010549599
FOR IDNO DKG1 DKG2 DKG2	ANNA A	
COEFFICIENTS = 0.0 = 0.0	ALT	
EFFI 0.0 0.0	-4	
PON CO	TAS	www.4444nnnnwwww.4444nnnnwwww.4440000000000
WEAP CFOR	BU EG	000000000000000000000000000000000000000



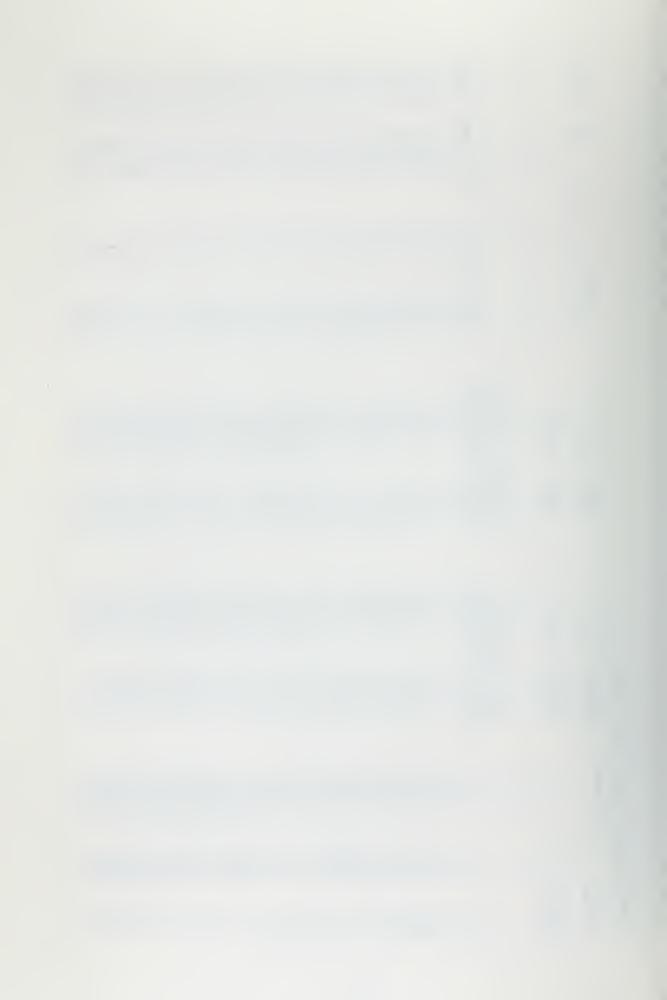


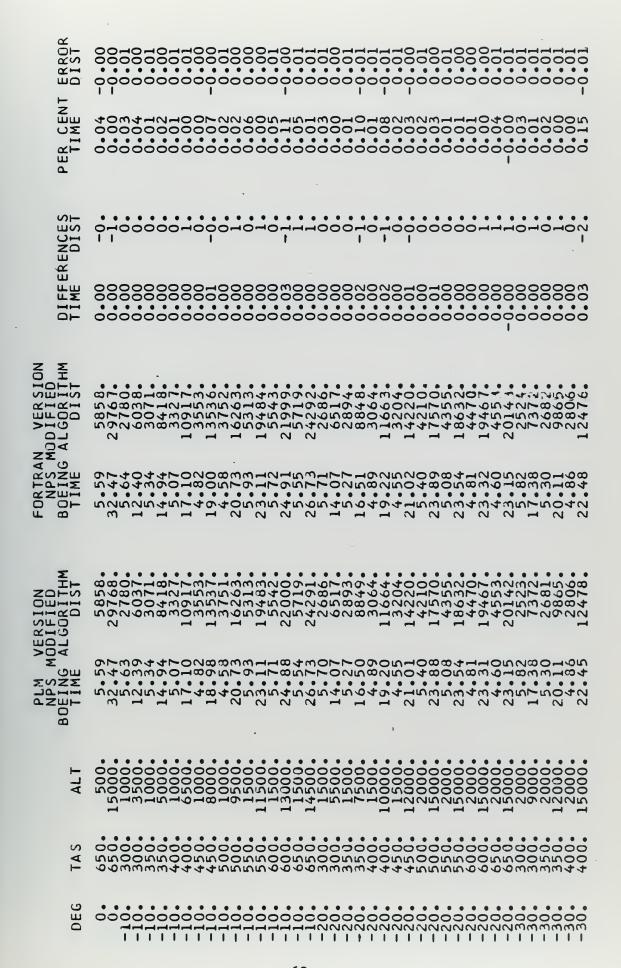


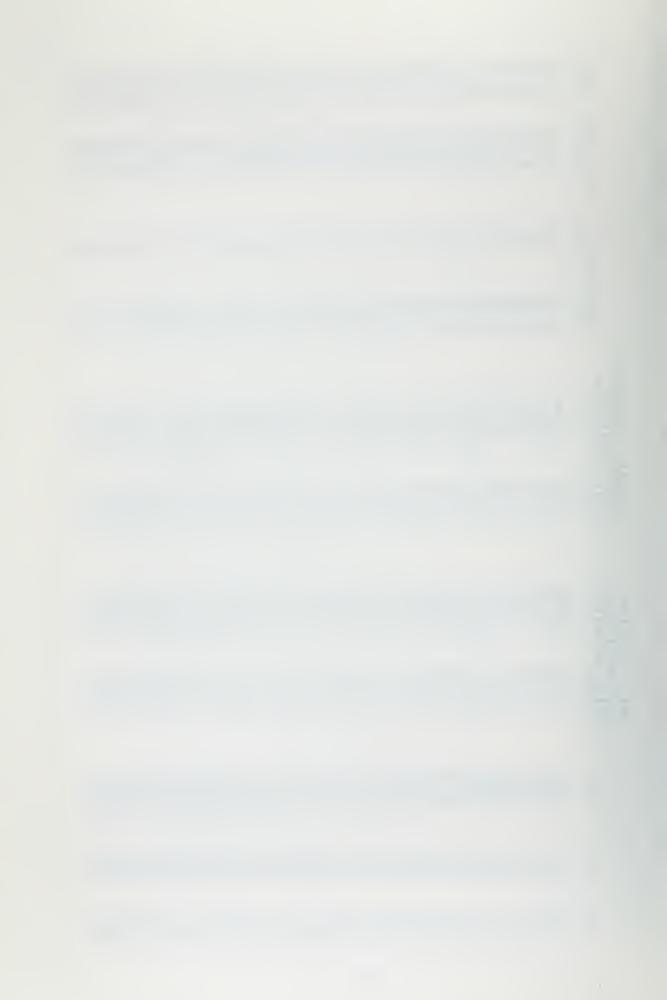
ERROR DIST	
PER CENT TIME	00000000000000000000000000000000000000
RENCES DIST	010000000000000000000000000000000000000
DIFFE	
MODIFIED ALGORITHM DIST	1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
FORTRAP NPS N BOEING TIME	2 2 2 1 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
RSION IFIED GORITHM DIST	114 1173 1173 1103 1103 1103 1103 1103 1103
PLM VE NPS MOD BOEING AL	2 2 2 2 1 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2
ALT	4 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
TAS	NNWWWWW4444NNNWWWWW4444NNNNNNNNNOONNOONN
DEG	11111111111111111111111111111111111111



0 • 0 = 0 • 0 0 • SL = 0 • 0	PER CENT ERROR TIME DIST	00000000000000000000000000000000000000
VWUZ FN =	DIFFERENCES TIME DIST	
DM1 = 0.0 DM2 = 0.0 VE = 0.0 DTI = 3.00	FORTRAN VERSION NPS MODIFIED BOEING ALGORITHM TIME DIST	16.76 17.32 17.32 17.32 17.32 10.27 10.27 10.27 10.27 10.27 10.27 10.97 11.64 11
FOR IDNO 9 DKG1 = 0.0 DKG2 = 0.0 IREF = 1 DMAX = 5.00	PLM VERSION NPS MODIFIED BOEING ALGORITHM TIME DIST	8.93 16.75 17.32 17.32 18.93 18.959 18.966 18.966 18.966 18.666 18.666 18.666 18.666 18.666 18.666 18.666 18.966 18.6666 18.
WEAPON COEFFICIENTS CFORM1 = 2.0639992 CFORM2 = 0.0 ITYPE = -1 IBOTH = 1	DEG TAS ALT	00000000000000000000000000000000000000



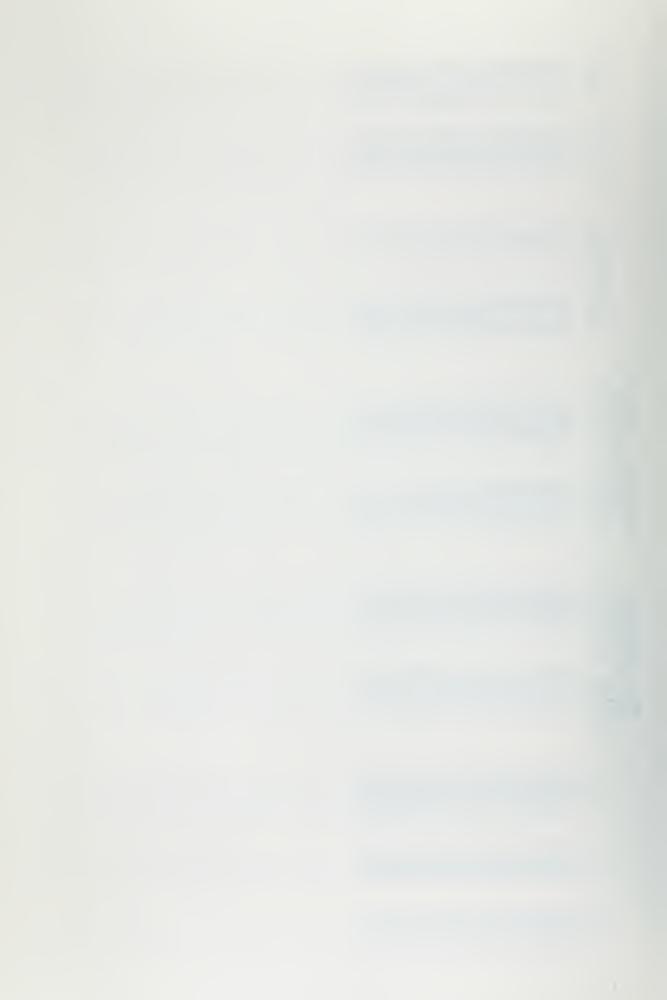




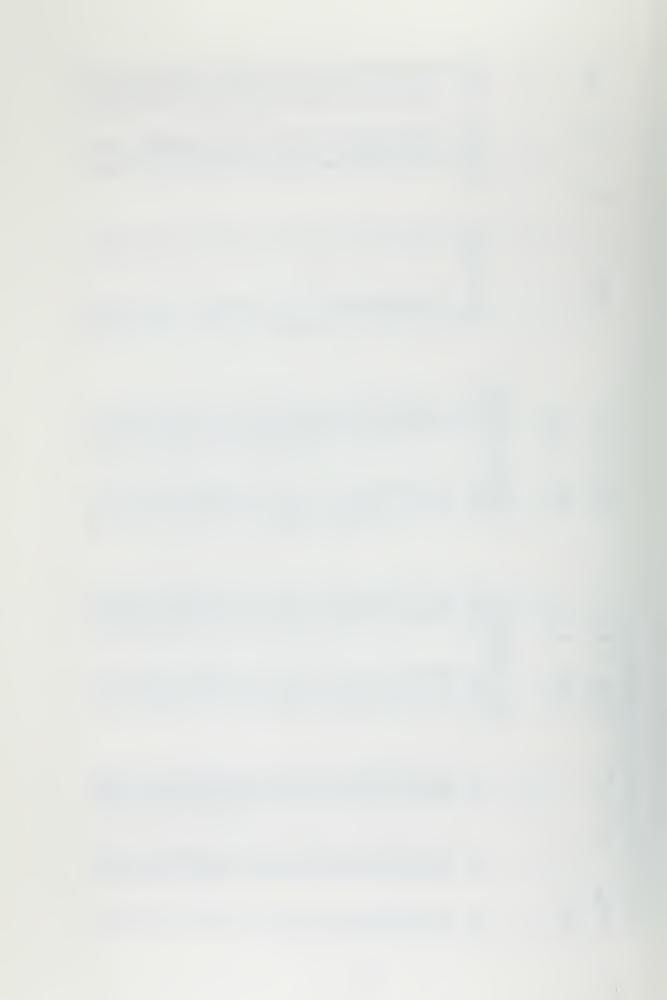
T ERROR DIST	
PER CEN	00000000000000000000000000000000000000
ER ENCES DIST	0,0000000000000000000000000000000000000
DIFFI	©0000000000000000000000000000000000000
MODIFIED ALGURITHM	1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
FORTRA NPS BOEING TIME	2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
ERSION DIFIED LGORITHM DIST	1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
PLM V NPS MO BOEING A	2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
ALT	\\ \text{\alpha}
TAS	44MNNN000000000000000000000000000000000
DEG	

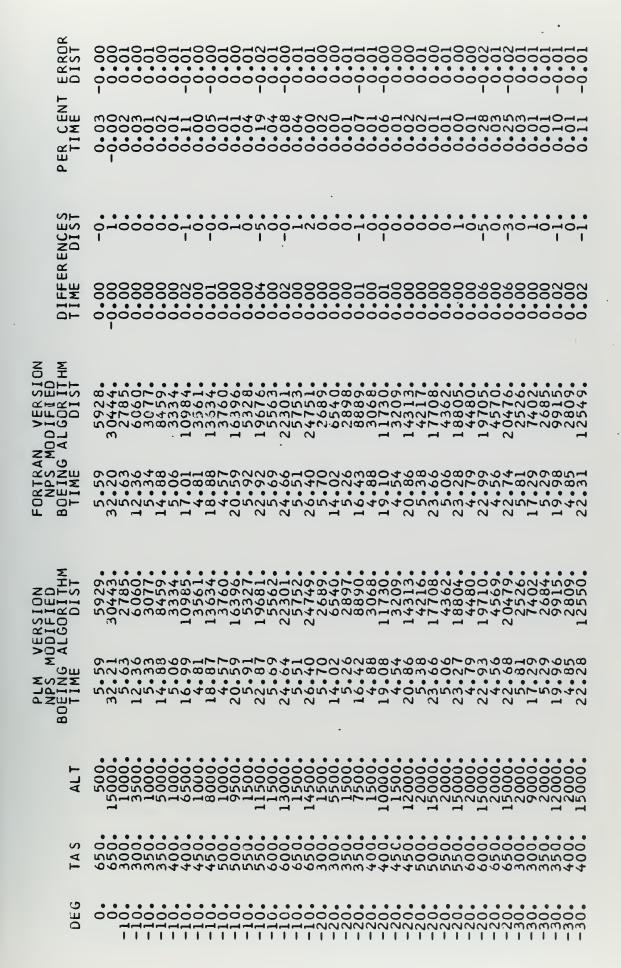


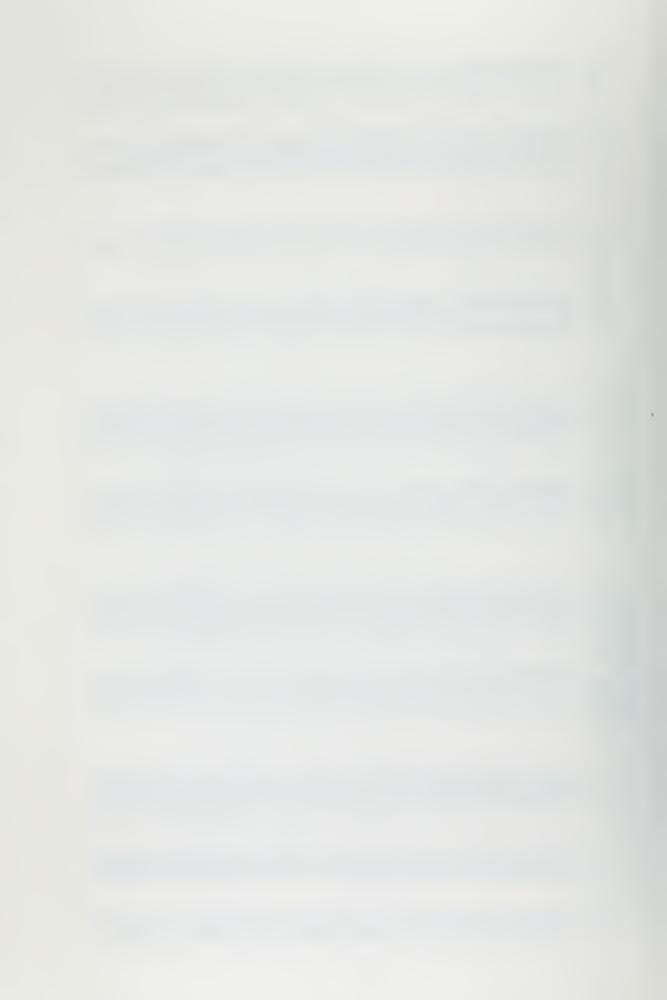
PER CENT ERROR TIME DIST	00000000000000000000000000000000000000
ER ENCES DIST	##000000000000000000000000000000000000
M DIFF	000000000000000000000000000000000000000
MODIFIED ALGORITHM DIST	100 000 000 000 000 000 000 000 000 000
FORTRA NPS BOEING TIME	1
ERSION DIFIED LGORITHM DIST	1102 1102 1102 1102 1003 1003 1003 1003
PLM NPS MO BOEING A	10000000000000000000000000000000000000
ALT	00000000000000000000000000000000000000
TAS	40wwww4444wwww4000 wwoowwoowwooww 0000000000
DEG	11111111111111111111111111111111111111



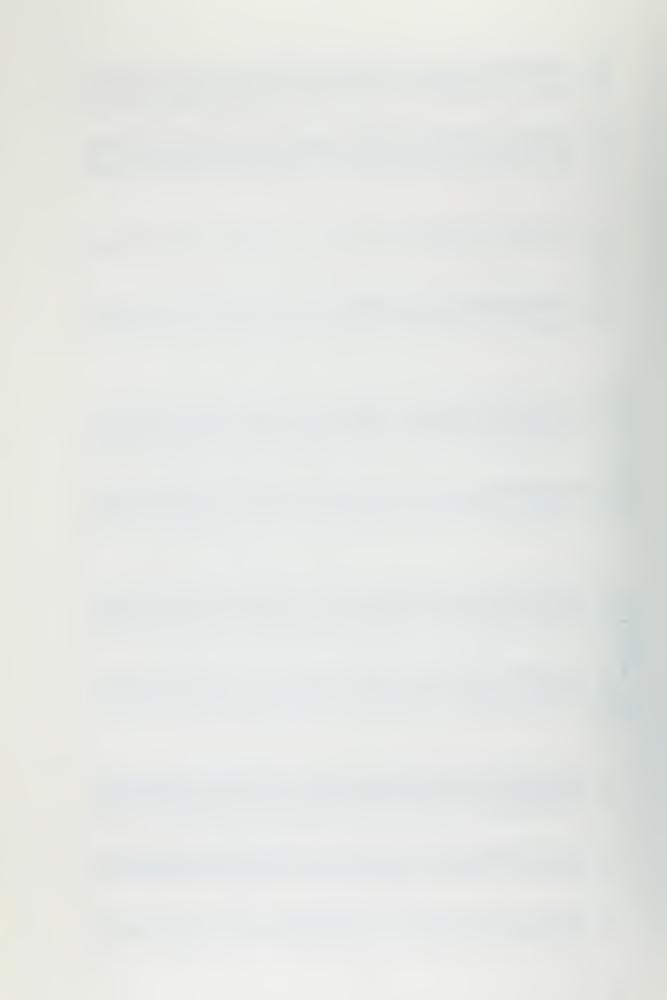
00	NA CONTRACTOR OF
0. 0. SL SL	AT 000000000000000000000000000000000000
33	ω ₊
= Zn	A D
>A EN	ПН 000000000000000000000000000000000000
	Z 5
00 00 00 00 00 00 00 00 00 00 00 00 00	AD A C C C C C C C C C C C C C C C C C C
DM1 : DM2 : DM2 : DTI :	В 1 1111111111
0	00000000000000000000000000000000000000
10	APURENSI GORILI 1130 200 200 200 200 200 200 200 2
FOR IDNO DKG1 DKG2 IREF	8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8
FICIENTS 4531993	AL A W W W W W W W W W W W W W W W W W W
PON COEFF RM1 = 1.4 RM2 = 0.0 PE = -1 TH = 1	 шиши4444ииии0000000ишиш4444ииии00 Ооииооииооииооииооииооииоо 000000000000000000000000000000000000
WEAPO CFORM CFORM ITYPE IBOTH	ш 000000000000000000000000000000000000



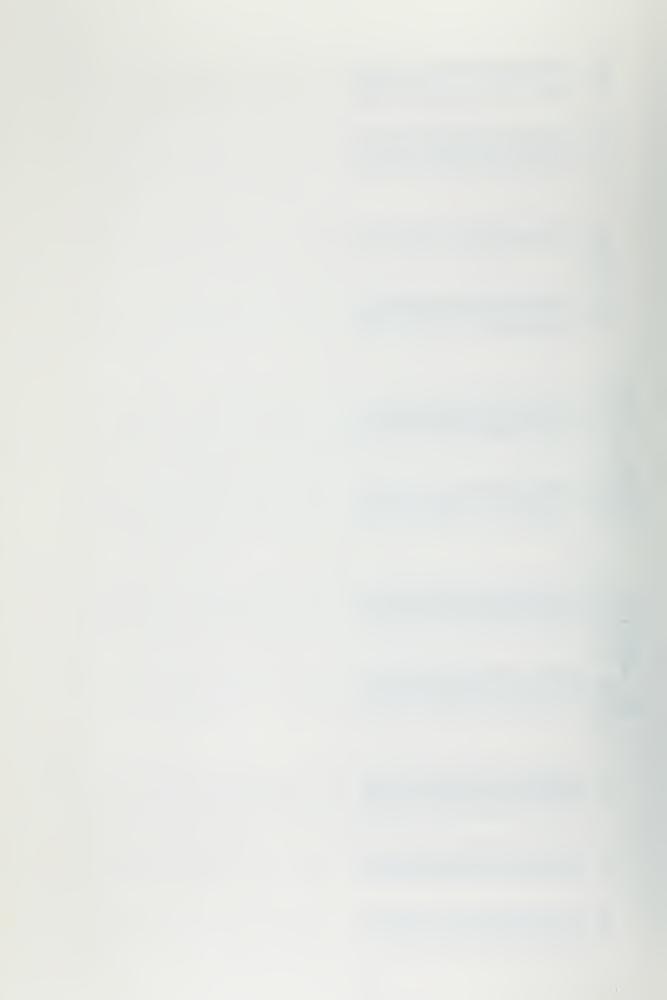




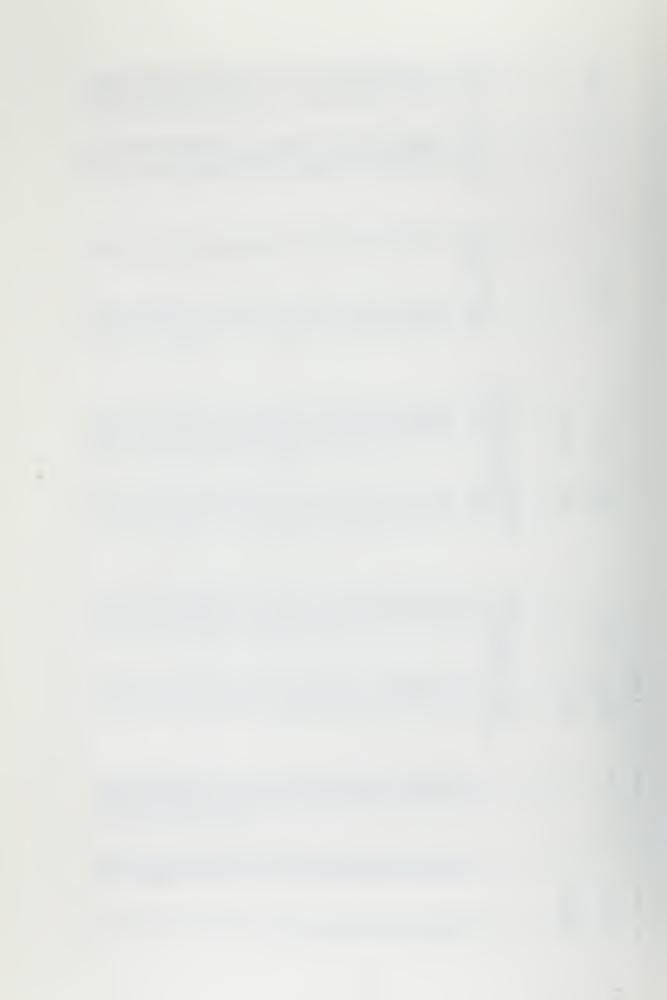
T ERROR DIST	
PER CENT	
RENCES	-0000
DIFFE	\$\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
NO HH	
N VERS MODIFIE ALGORI	1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
FORTRAL NPS BOEING TIME	0.0 2 0 1 1 2 2 2 1 1 2 2 2 1 1 1 2 2 2 1 1 1 1 2 2 2 1 1 1 1 2 2 2 1 1 1 1 2 2 2 1 1 1 1 2 2 2 1 1 1 1 2 2 2 1 1 1 1 2
ERSION DIFIED LGORITHM DIST	1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
PLM VE NPS MOE BOEING AL	1
ALT	\\ NUNUNUMUMUMUMUMUMUMUMUMUMUMUMUMUMUMUMUM
	44mmma909mmu4444mmm0009mmu4444mmm00 mv00mv00mv00mv00mv00mv00mv00mv00 00000000
DEG	

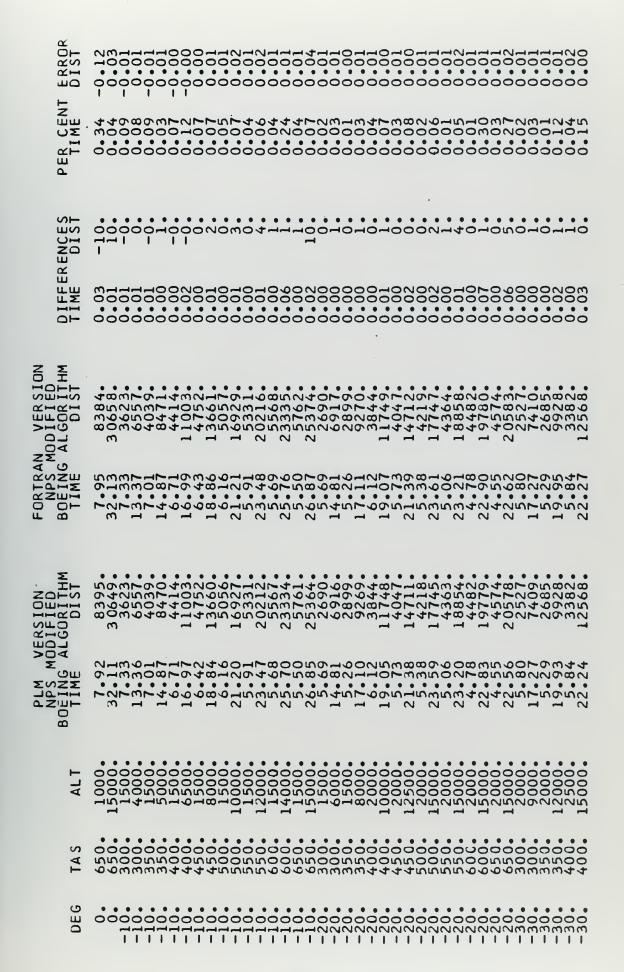


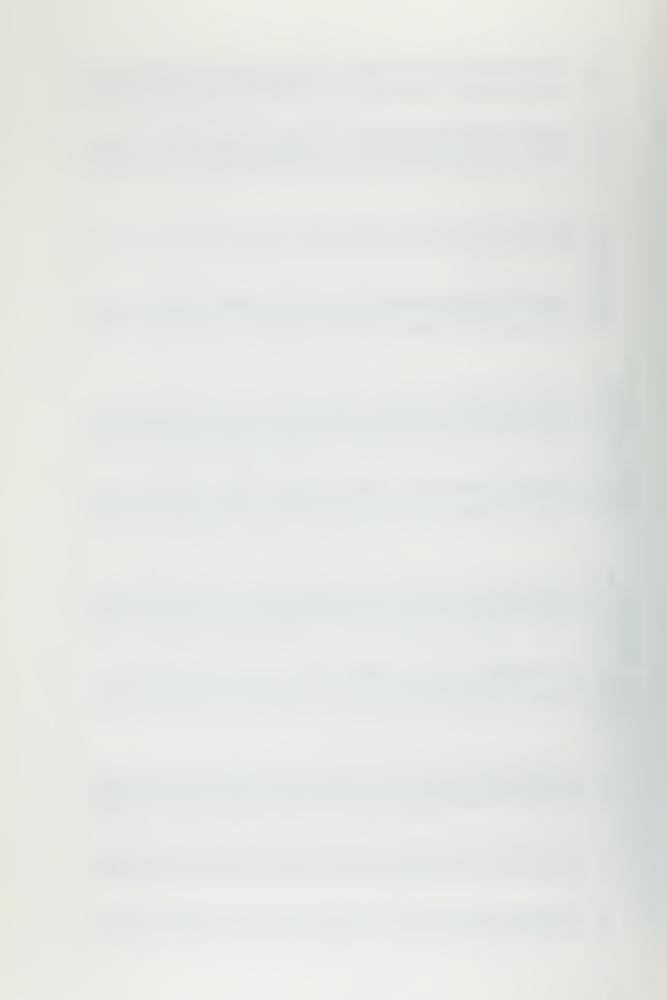
PER CENT ERROR TIME DIST	00000000000000000000000000000000000000
RENCES	
DIFFE	000000000000000000000000000000000000000
VERSION MODIFIED ALGORITHM DIST	1152 1172 1172 146 146 146 146 146 146 146 146 146 146
FORTRAN BOEING TIME	16
RSION IFIED GORITHM DIST	1112 1112 1112 1112 122 123 123 123 123
PLM VE NPS MOD BOEING AL	16.00 16.00 17.00 17.00 17.00 18.00 19.00 19.00 10.00
ALT	00000000000000000000000000000000000000
TAS	00000000000000000000000000000000000000
DEG	11111111111111111111111111111111111111



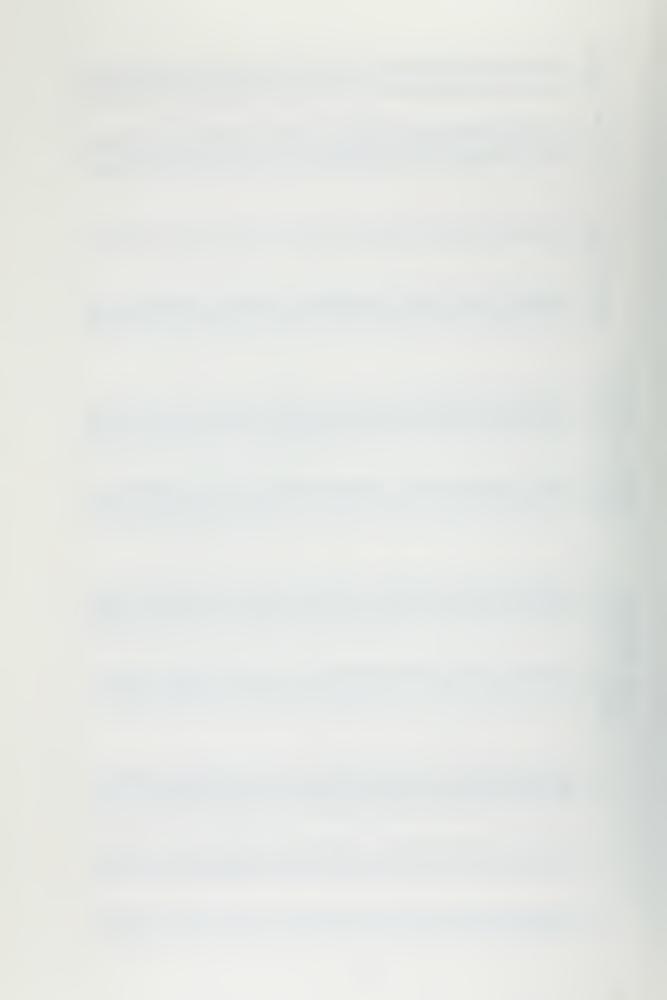
	0000		T ERROR DIST	
	DS SL		PER CENT	00000000000000000000000000000000000000
	00	•	•	
	= Zn		ER ENCES DIST	
	N N N		DIFF	000000000000000000000000000000000000000
			Ζ Σ	
	0000	0 • 0 • 1 • 00	MODIFIED ALGORITHM DIST	2 2 2 2 1 1 211111 1 1 2 2 2 2 2 2 2 2
	DM1 DM2	VE := DTI	FORTRA NPS BOEING TIME	8 9 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8
11	00	5.00	ERSION DIFIED LGORITHM DIST	2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
FOR IDNO	DKG1 = DKG2 =	IREF = DMAX =	PLM NPS MO TIME ATIME	12111111111111111111111111111111111111
FICIENTS	3430996		ALT	11111111111111111111111111111111111111
ON COEFF	M1 = 1. $M2 = 0$.	H H	TAS	
WEAPON	CFOR	I BOTH	DEG	000000000000000000000000000000000000000



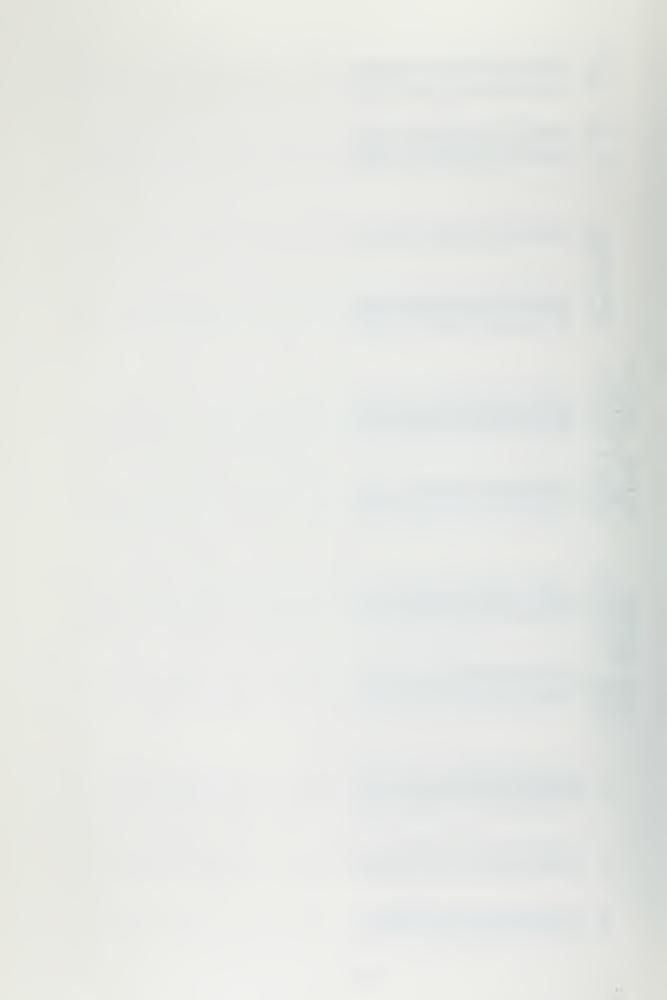




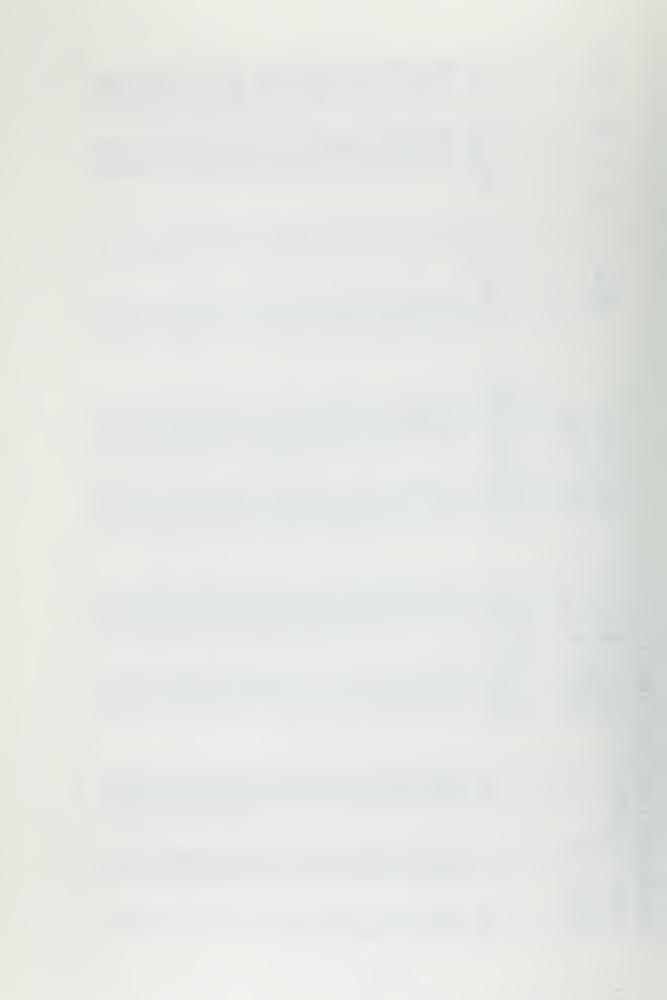
ERROR DIST	
PER CENT TIME	00000000000000000000000000000000000000
RENCES	0-000-4
DIFFE	
SION ED ITHM ST	######################################
MODIFI ALGOR	1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0
FORTRAI NPS BOEING TIME	1
RS I ON I FIED GORITHM DIST	1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
PLM VE NPS MUD BOEING AL TIME	1
ALT	
TAS	
DEG	



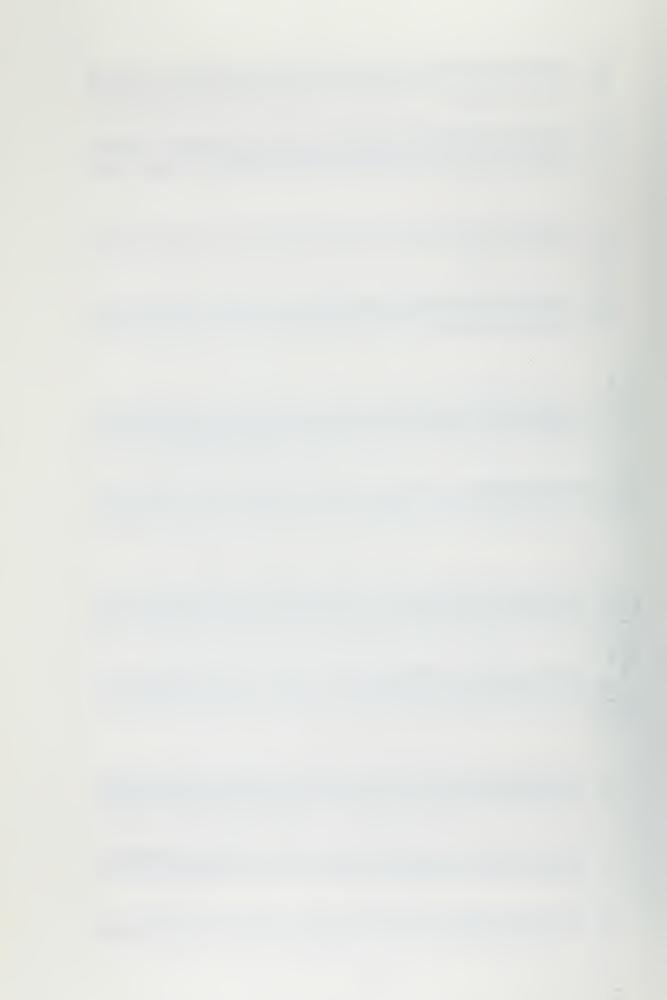
ENT ERROR E DIST	00000000000000000000000000000000000000
PER CE TIME	
ERENCES DIST	000000000000000000000000000000000000000
DI FF	000000000000000000000000000000000000000
NODIFIED NACORITHM	1 2464696000000000000000000000000000000000
FORTRAI BOEING TIME	2 1 1 1 2 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2
ERSION DIFIED LGORITHM DIST	1122223 1122223 192223 193223 19323 19323 19323 1933 193
PLM V NPS MO BOEING A	100-100 100
ALT	00000000000000000000000000000000000000
TAS	00 00 00 00 00 00 00 00 00 00 00 00 00
DEG	449999999999999999999999999999999999999

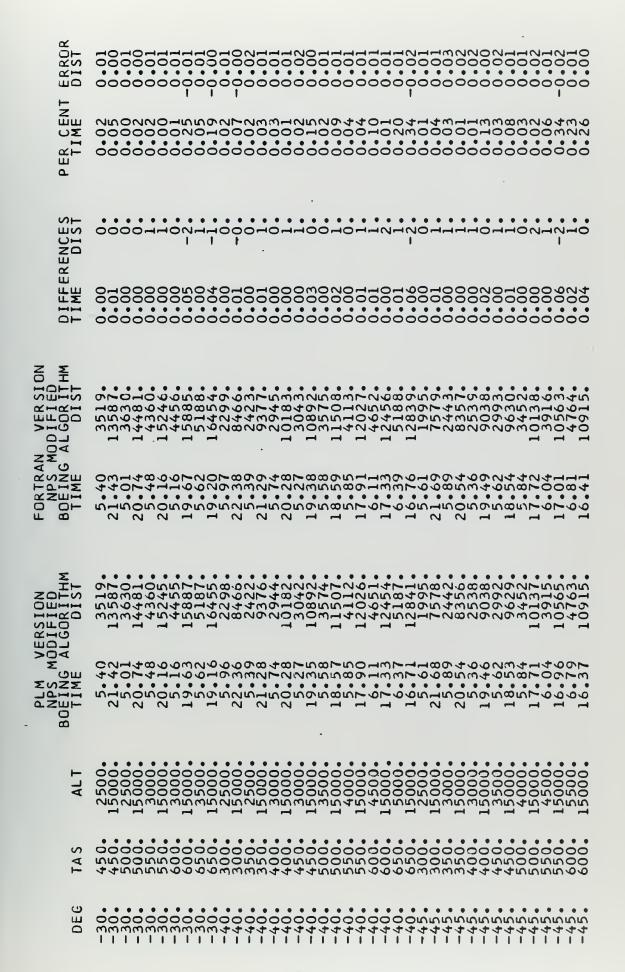


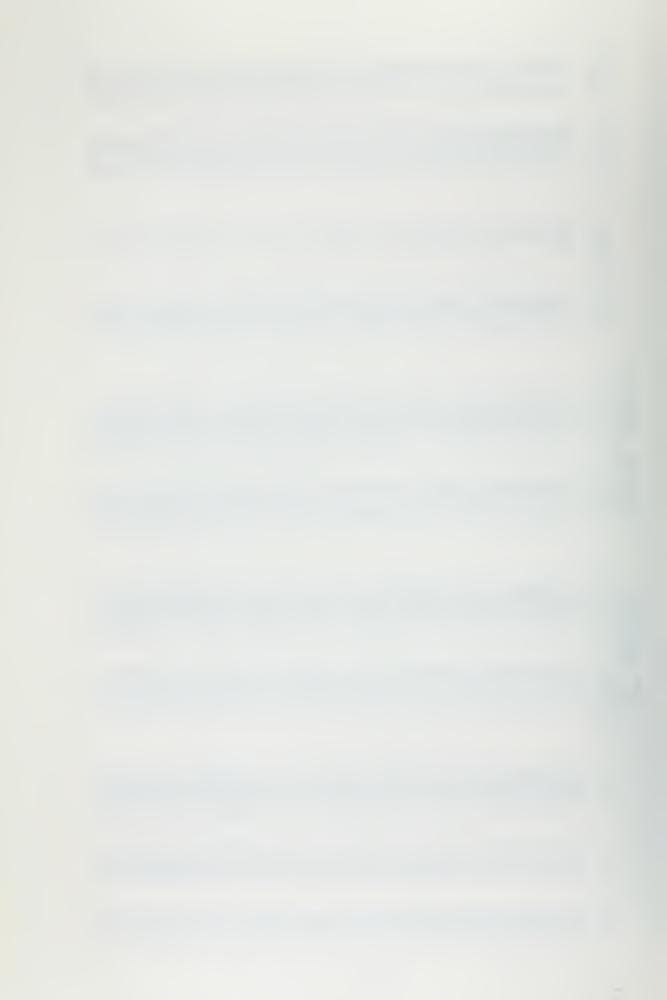
00000	T ERR DIS	
DS SL	I ACE	00000000000000000000000000000000000000
00		
= Zn	ERENCES DIST	
∑Z >L	TH C	
	N F	
00000	MODIFIED ALGORIT DIST	2 2 2 2 2 1 1 2 2 2 2 2 2 2 2 2 2 2 2 2
DM1 DM2 VE =	AZUH TOTE	2 3 3 3 3 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5
12 = 0.0 = 0.0 = 1.	NOTI N	2 2 2 2 2 1 1 2 1 1 1 2 2 2 2 2 2 2 2 2
FOR IDNO DKG1 DKG2 IREF DMAX	ENZE Om o	3 3 3 3 000000000000000000000000000000
COEFFICIENTS = 1.2099991 = 0.0 -1	LT	
APON COEF ORM1 = 1. ORM2 = 0. YPE = -1	AS	
WEAP CFOR CFOR ITYP IBOT		000000000000000000000000000000000000000



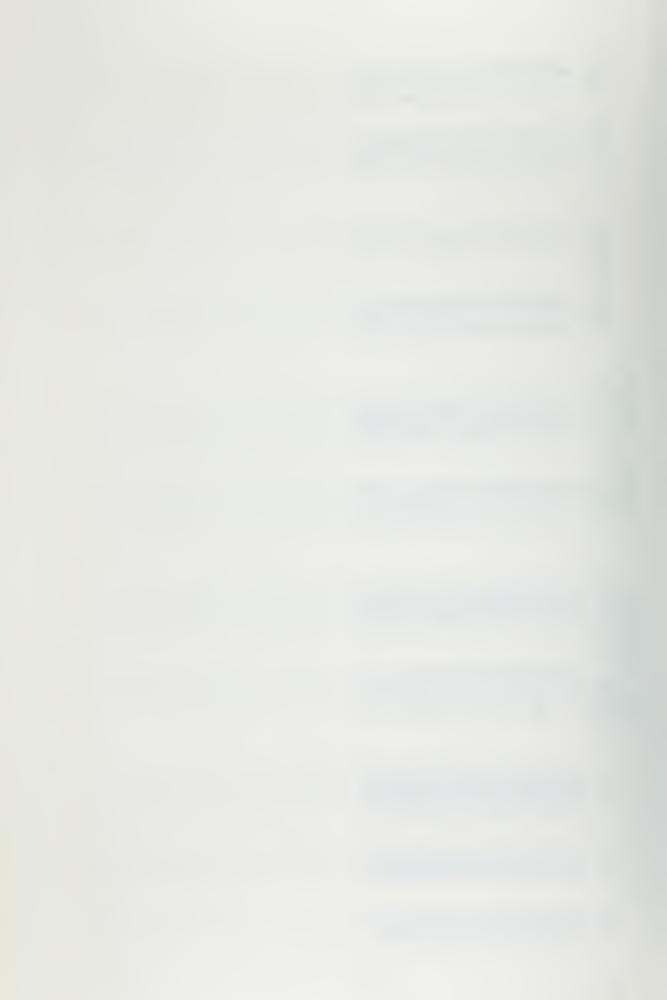
IT ERROR DIST	
P ER CEN	
ER ENCES DIST	1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
DIFF	
MODIFIED ALGORITHM	2 2 1 1 1 2 2 2 1 1 1 1 2 2 2 2 1 1 1 1
FORTRA NPS BOEING TIME	2 1 1 1 2 2 2 2 1 1 1 1 2 2 2 2 1 1 1 2 2 2 2 2 1 1 1 2 2 2 2 2 1 1 1 2 2 2 2 2 1 1 1 2 2 2 2 2 2 1 1 1 2 2 2 2 2 2 2 1 1 1 2
VERSION ODIFIED ALGORITHM DIST	2 2 2 1 1 1 1 4 8 4 6 9 8 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9
PLM NPS BOEING TIME	2 2 2 2 2 1 1 2 2 2 2 2 1 1 2 2 2 2 2 1 1 2 2 2 2 2 1 1 2 2 2 2 2 2 1 1 1 2 2 2 2 2 2 1 1 1 2 2 2 2 2 2 1 1 1 2 2 2 2 2 2 2 1 1 1 2
ALT	
	OWWWWWA****************************
DEG	



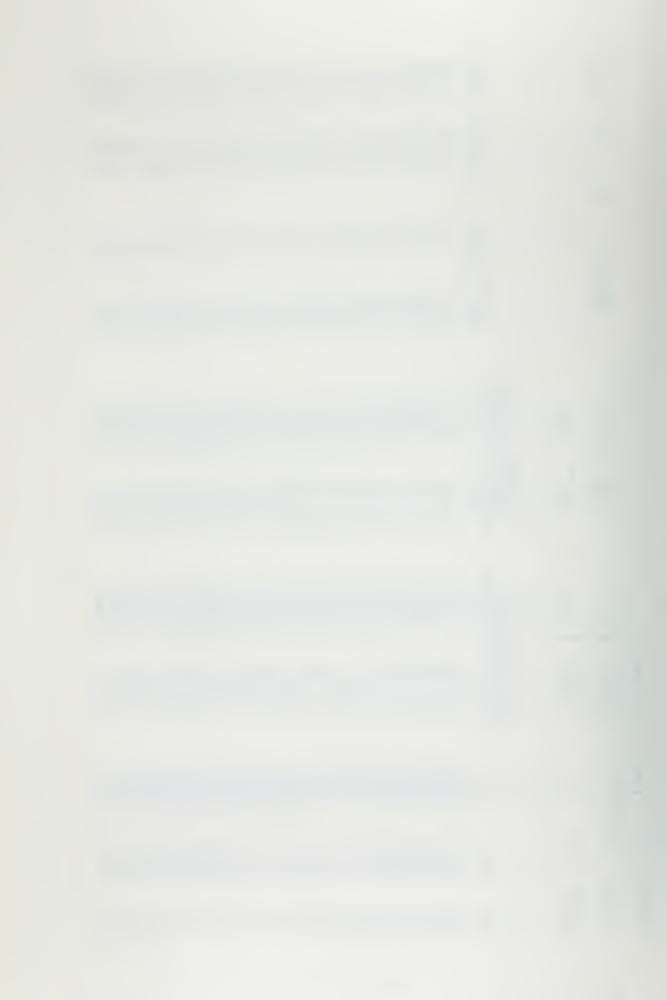


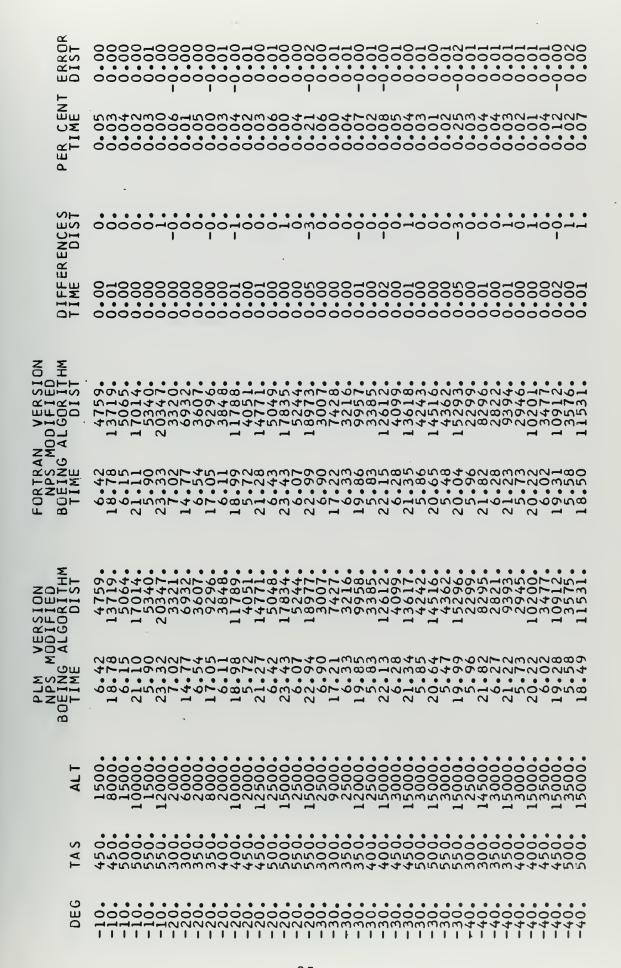


	•
00000000000000000000000000000000000000	t
000000000000000000000000000000000000000	•
1122 1122 1122 1222 1222 1222 1222 122	7
2017- 100- 100- 100- 100- 100- 100- 100-	7
11 11 12 12 12 12 12 12 12 12 13 13 13 13 13 13 13 13 13 13 13 13 13	
1	
8 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	20004
00000000000000000000000000000000000000	5
11111111111111111111111111111111111111	-60
	650. 6500. 6.97 15226. 7.00 15226. 0.03 0.03 0.037 0.



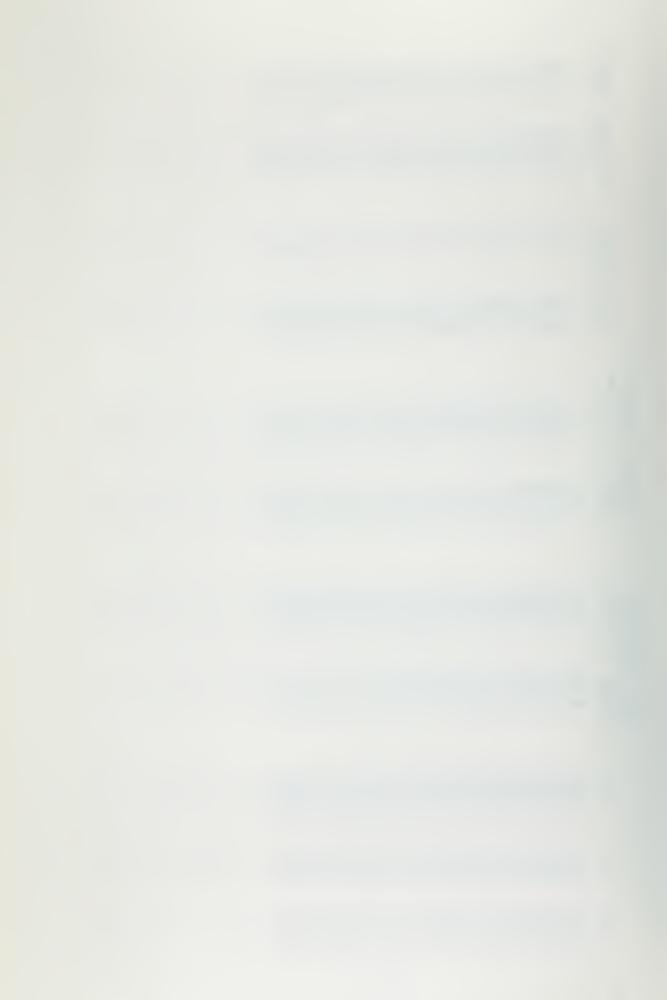
	S = 0.0		NT ERROR DIST	
	0.0		PER CET	00000000000000000000000000000000000000
	= 701		ERENCES	0-00-040-00-00-00-00-00-00-00-00-00-00-0
	>¤ ∑X		OIFF TIME	000000000000000000000000000000000000000
	0.0	3.00	MODIFIED ALGORITHM	2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
	DM1 DM2	VE B	FORTRA NPS BOEING TIME	11111111111111111111111111111111111111
13	0.0	= 5.00	VERSION ODIFIED ALGORITHM DIST	2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
S FOR IDNO	DKG1 DKG2	IREF DMAX	PLM NPS NPS TIME	11111111111111111111111111111111111111
COEFFICIENTS	00000000		ALT	6 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
	tM1 = 1 tM2 = 0	## ## ##	TAS	www.4444\n\n\n\w\w\w\w\\\\\\\\\\\\\\\\\\
WEAPON	CFOR	ITYPIBOT	DEG	



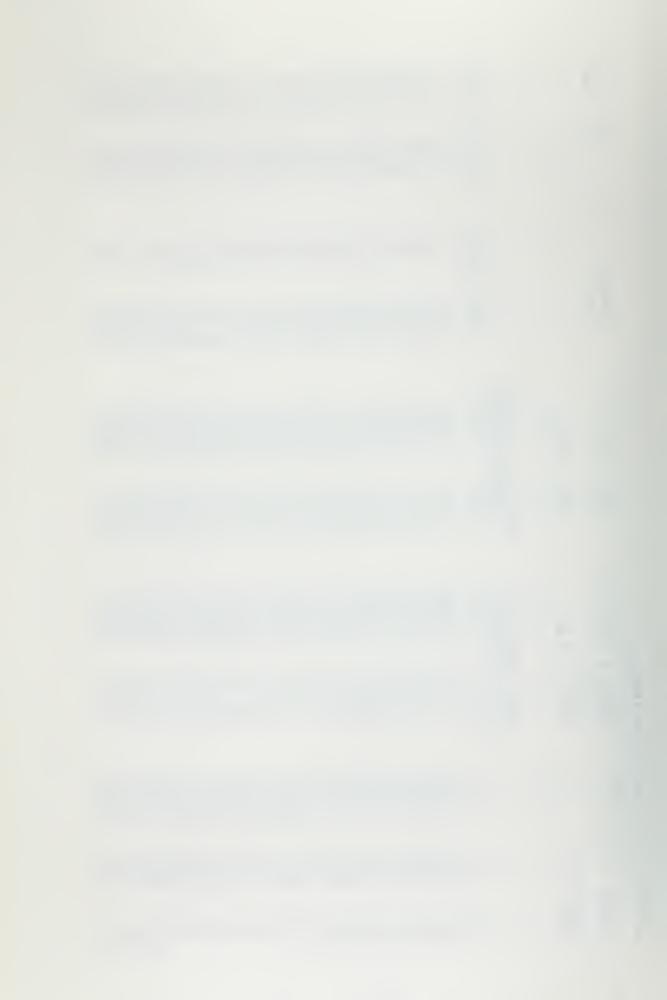




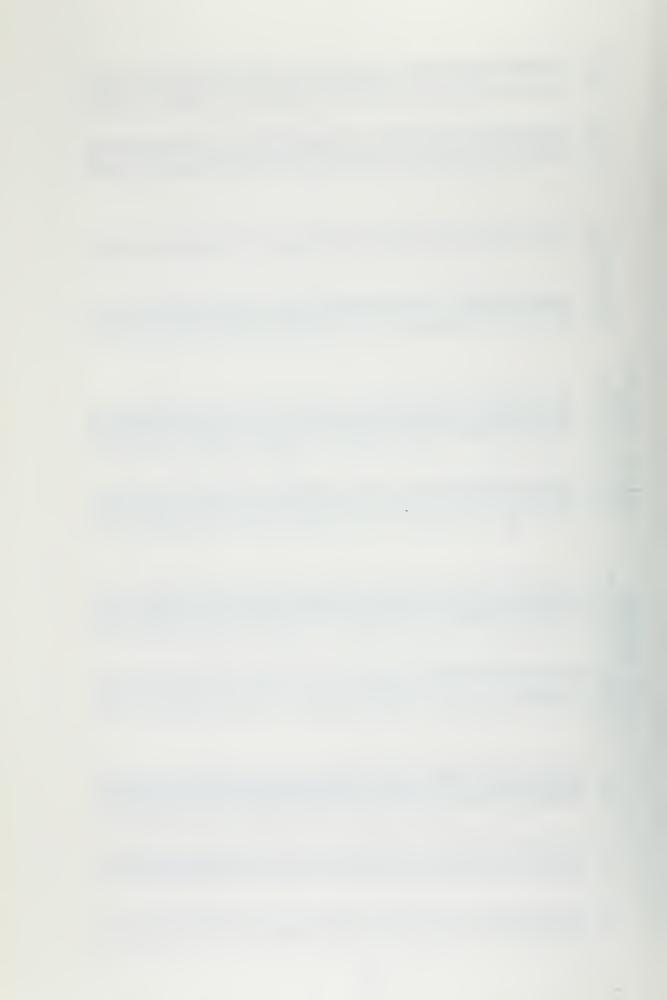
ERROR DIST	
PER CENT TIME	00000000000000000000000000000000000000
ENCES	01011011011011010000010001
DIFFER	
VERSION MODIFIED ALGORITHM DIST	1 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
FORTRAN NPS M BOEING TIME	1 1 2 2 1 1 2 2 2 1 2 2 2 1 2 2 2 2 2 2
ERSION DIFIED LGORITHM DIST	1
PLM V NPS MO BOEING A	201020 1050050 105000 105000 105000 105000 105000 105000 105000 105000 105000 105000 105000 105000 105000 105000 105000 105000 105000 105000 1
ALT	4 WWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWW
TAS	
DEG	1



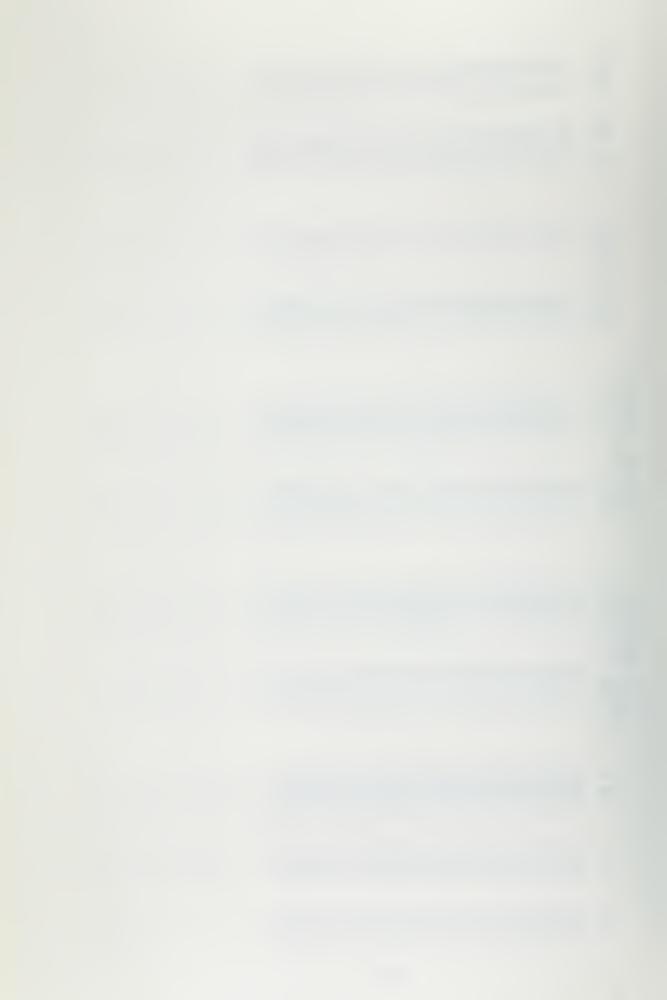
	0.0		T ERROR DIST	
	DS SL		PER CENT TIME	00000000000000000000000000000000000000
	00			
	= 71		ERENCES DIST	
	N N N		DIFFE TIME	000000000000000000000000000000000000000
			N M	
	00000	9.00	N VERSIC MODIFIED ALGORITH DIST	2 2 2 1 1 1 1 1 1 1 69 5 8 4 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8
	DM1 ==	VE = DTI	FORTRAN NPS 1 BOEING TIME	1 1 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
	30			
ON COEFFICIENTS FOR IONO 14	001223	5.00	ERSION DIFIED LGORITHM DIST	1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
	1 = 2	 	>0 >0 >0 1	00000000000000000000000000000000000000
	DKG	IRE	PLM NPS BOEIN	
	M1 = 3.1199999 M2 = 0.0		ALT	
			TAS	www.4444nnnnwww.4444nnnnwww.444
WEAPON	CFOR	ITYPE	DEG	000000000000000000000000000000000000000



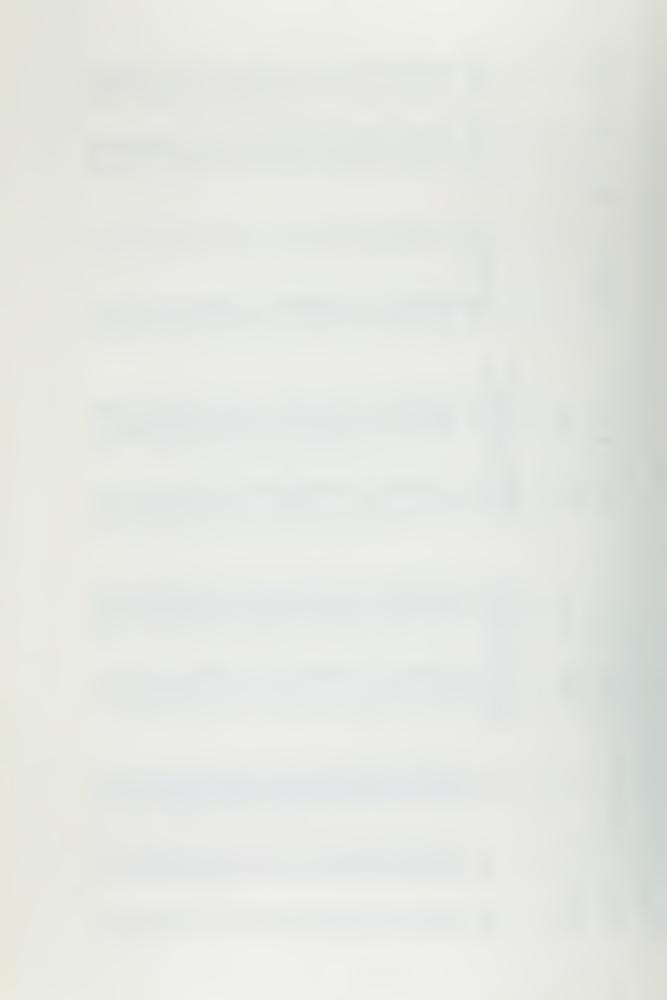
T ERROR DIST		•
PER CENT	00000000000000000000000000000000000000	7
RENCES	0-1000000000000000000000000000000000000	
DIFFE		•
MODIFIED ALGORITHM DIST	1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	001
FORTRA NPS BOEING TIME	1 0 0 1 1 1 0 0 0 1 0 0 0 0 0 0 0 0 0 0	•
ERSION DIFIED LGORITHM DIST	1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	0
PLM NPS MOING ALING ALIME		•
ALT		
TAS	44NNNNWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWW	
DEG		F

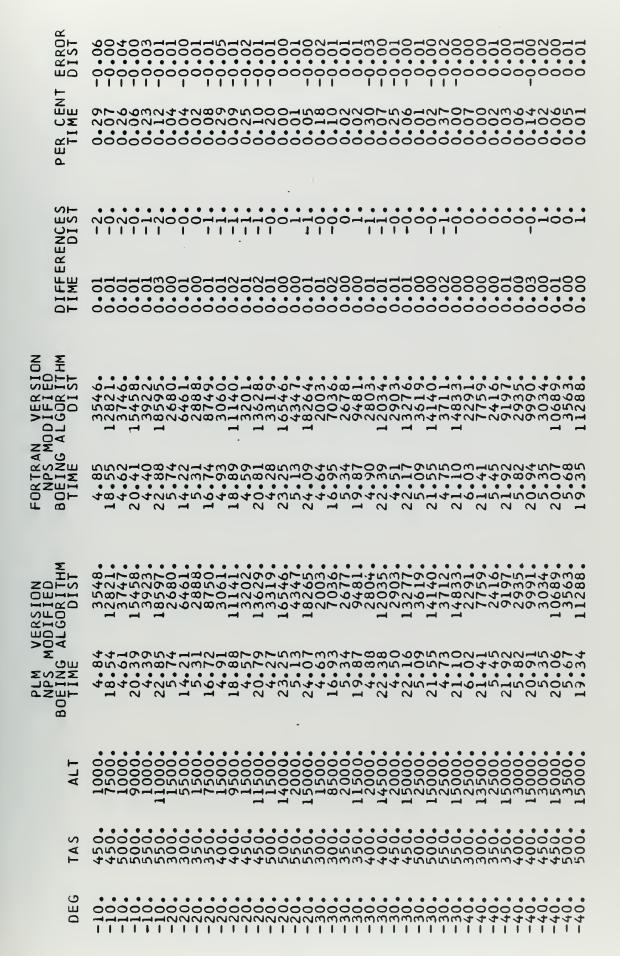


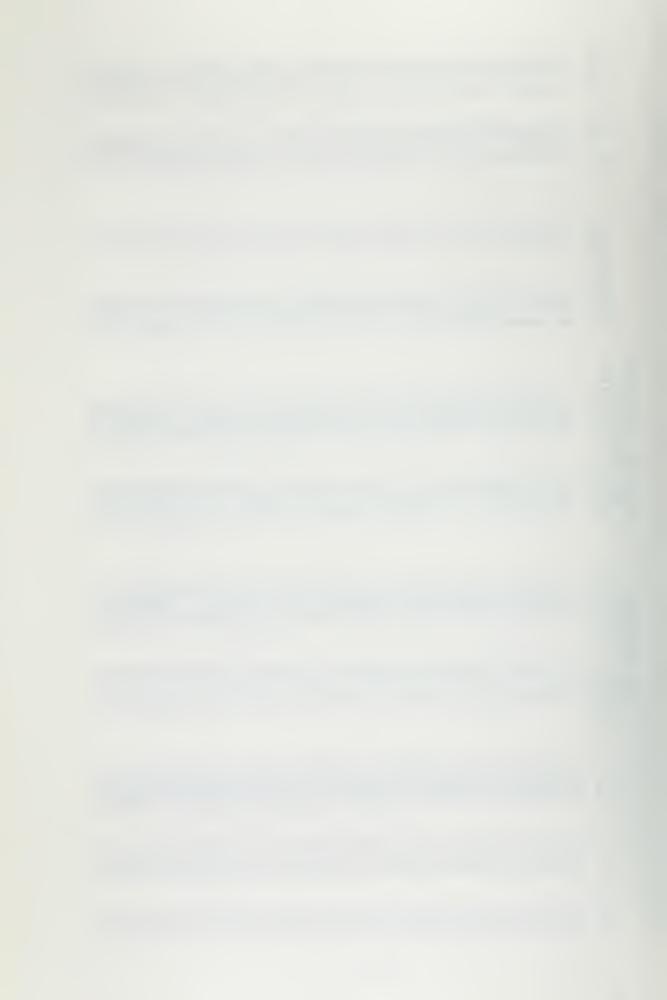
œ	
T ERROF DIST	000000000000000000000000000000000000000
P ER CEN	00000000000000000000000000000000000000
RENCES	
DIFFE	######################################
N M	
VERSI ODIFIED ALGURIT	1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
FORTRAN NPS M BOEING TIME	1 1 1 1 1 2 1 1 1 2 2 1 1 1 1 2 2 2 1 2
RS ION IFIED GORITHM DIST	1 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
MOD ACC	00mm24gm2gga6g0ggm2gga4g
PLM NPS BOEING TIME	
ALT	12000000000000000000000000000000000000
TAS	<i>ონოოლო 444 დ დ დ 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6</i>
DEG	



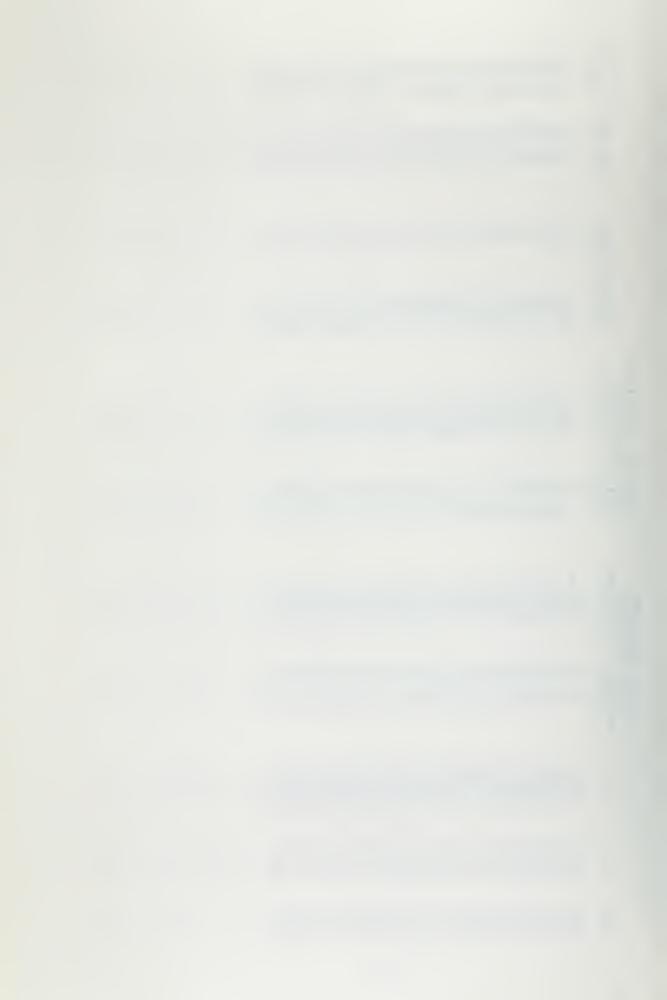
	0000		IT ERROR DIST	000000000000000000000000000000000000000
	0S		PER CEN	00000000000000000000000000000000000000
	00			
	= 71		ERENCES DIST	N-00-0m-00-00-00-0-00000000000000000000
	⊃ EN SN		DIFFE	
			Z ¥	•
	0.0	= 2.00	N VERSIO MODIFIED ALGORITH DIST	2 2 2 1 1 1 1 1 1 1 1 1 1 1 2 2 2 2 2 2
	DM1 =	VE = DTI =	FORTRAN NPS N BOEING TIME	1
			Σ	• • • • • • • • • • • • • • • • • • • •
51	00.0	3.00	ERSION DIFIED LGORITH DIST	2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
1 0	1 = 2	 L×	M A OF	00000000000000000000000000000000000000
FOR IDNO	DKG DKG	IRE! DMA)	BOEIN TIME	п п п п п п п п п п п п п п п п п п п
FICIENTS	0		ALT	## ## ## ## ## ## ## ## ## ## ## ## ##
ON COEFF	1 = 3. 2 = 0.		TAS	wwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwww
WEAPON	CFORM	I TYPE I BOTH	DEG	



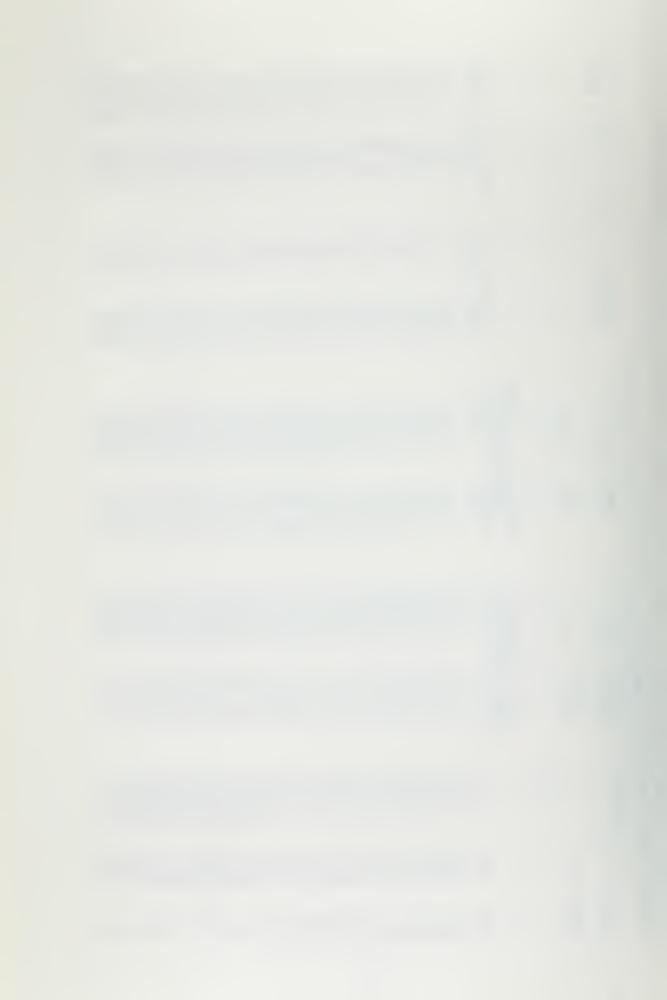




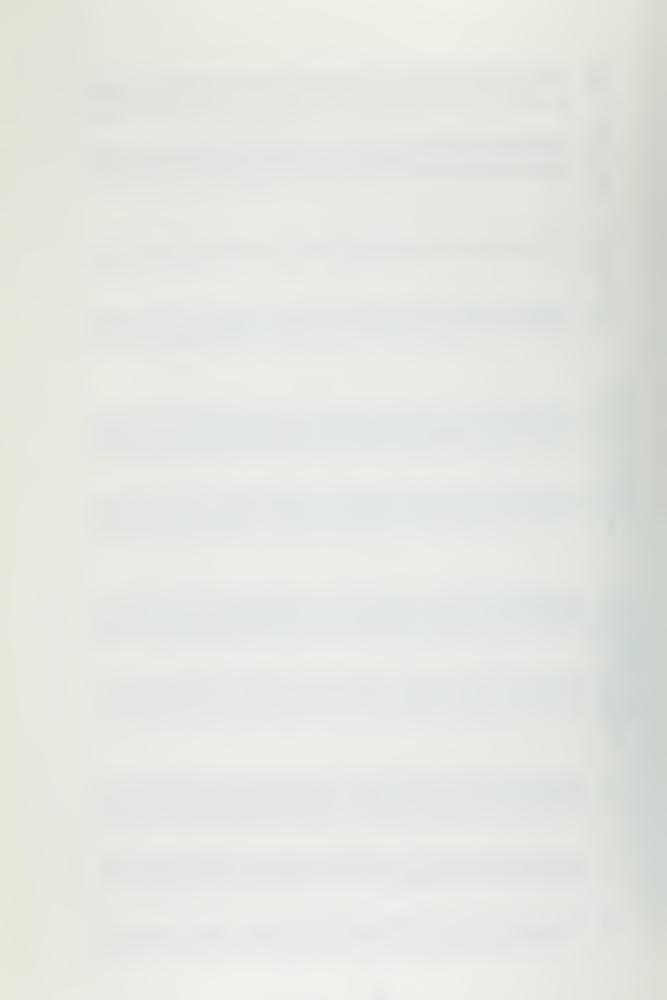
T ERROR DIST	000000000000000000000000000000000000000
PER CEN	
ENCES	0-00
OIFFER TIME	
VERSION MODIFIED ALGORITHM DIST	1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
FORTRAN NPS M BOEING TIME	2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
IFIED GORITHM DIST	11 12 13 14 10 10 10 10 10 10 10 10 10 10
PLM VE NPS MOD BCEING AL	2 2 2 2 1 1 2 1 1 2 2 2 2 2 2 2 2 2 2 2
ALT	4 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
TAS	NNWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWW
DEG	



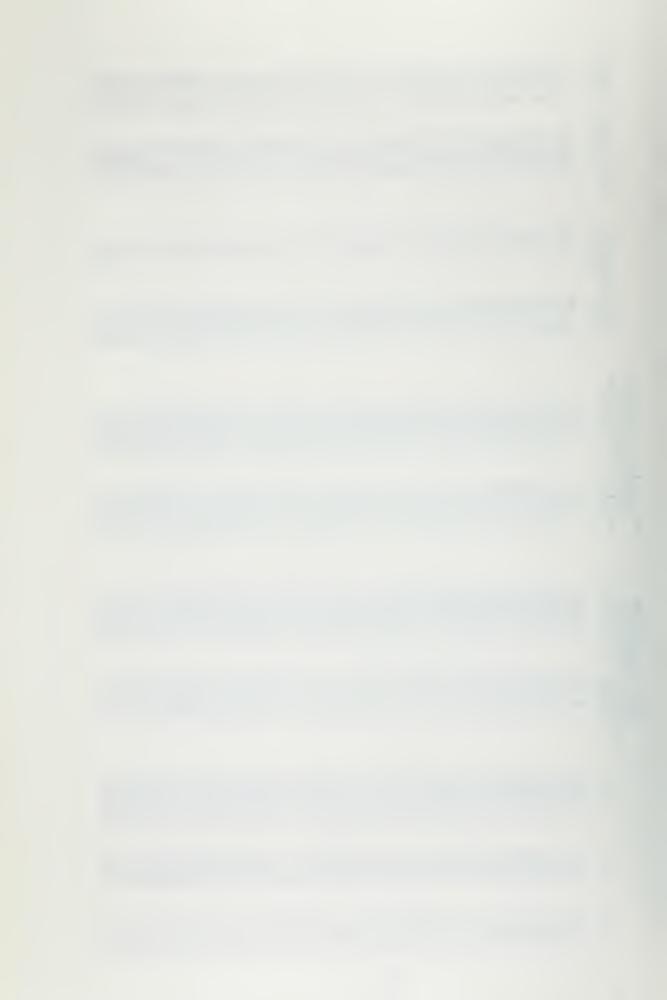
00000	NT ERROR DIST	
DS SL	PER CEN TIME	000000000000000000000000000000000000000
00		
= =	ERENCES DIST	
>r ××	DIFF TIME	000000000000000000000000000000000000000
	N F	
00000	N VERSI MODIFIED ALGORIT DIST	2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
DM1 DM2 VE =	FORTRA NPS BOEING TIME	21211111111111111111111111111111111111
16 = 0.0 = 0.0 = 1.00	VERSION ODIFIED ALGORITHM DIST	2 2 2 2 2 1 1 2 2 2 2 2 2 2 2 2 2 2 2 2
FOR IDNO DKG1 = DKG2 = IREF =	PLM NPS MC BOEING A	8 3 2 1211111111111111 1 8 8 8 8 8 8 8 8 8
COEFFICIENTS = 1.6049995 = 0.0 -1	ALT	
HU WW O	TAS	
WEAP CFOR CFOR ITYP IBOTE	DEG	000000000000000000000000000000000000000



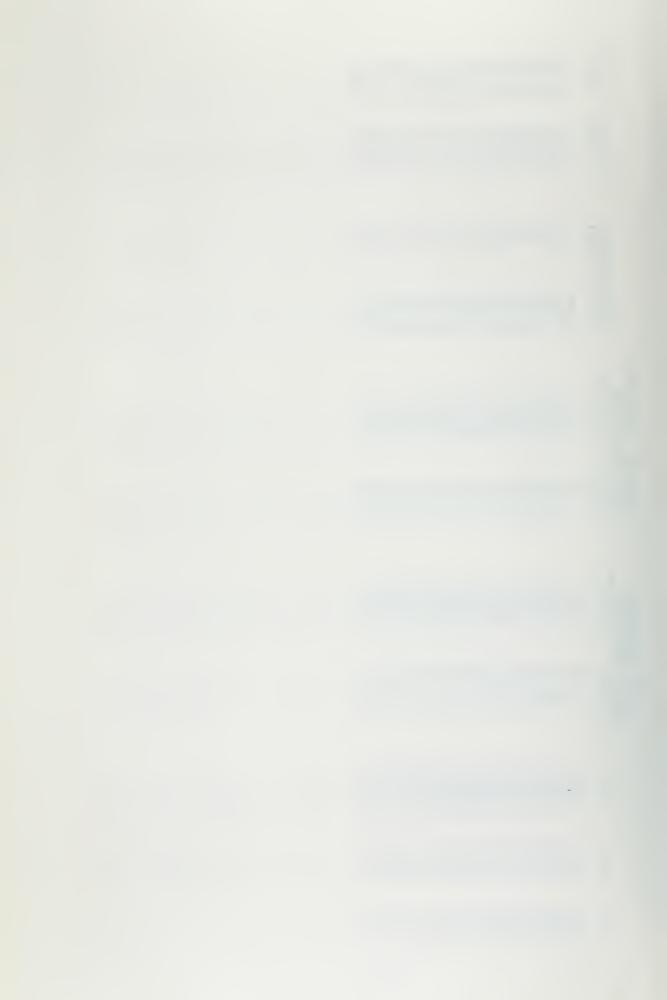
. ERROR DIST	
PER CENT	
RENCES	
DIFFE	
ADDIFIED ALGORITHM DIST	2 2 2 1 1 1 2 2 2 2 1 1 1 1 2 2 2 2 2 1 1 1 1 2
FORTRAN NPS M BOEING TIME	2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
DIFIED LGORITHM DIST	2 2 2 1 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
PLM NPS MOI BOEING A	2 1 1 2 2 2 2 2 1 1 1 2 2 2 2 2 1 1 2
ALT	12220000000000000000000000000000000000
	$\begin{array}{c} \delta \phi u u u u u 4 4 4 4 \ell \ell$
DEG	



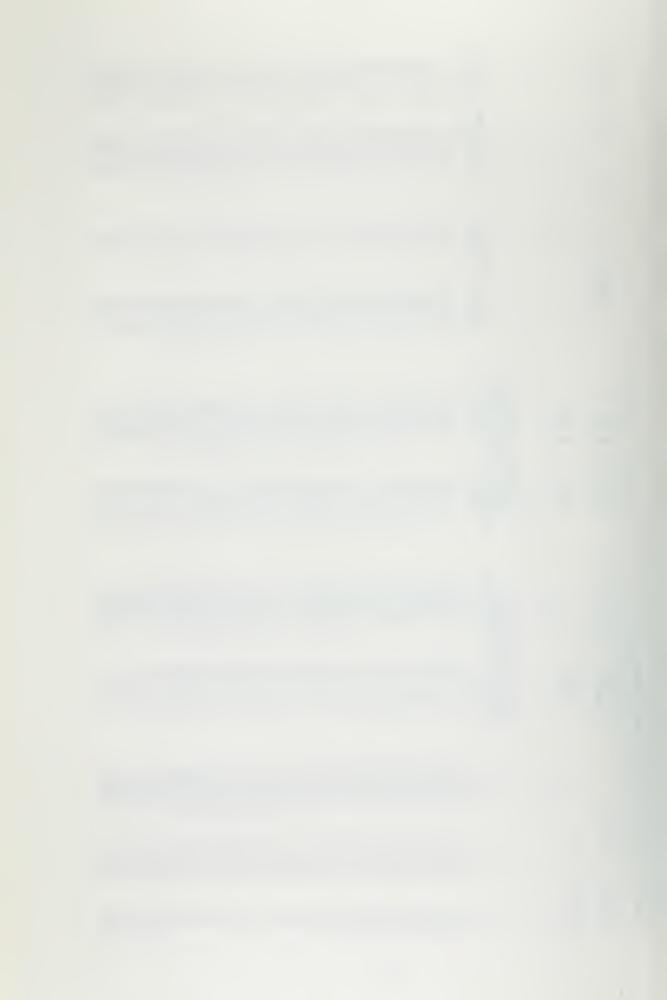
α.	
ERRO DIST	
ER CENT	00000000000000000000000000000000000000
9	
RENCES DIST	
DIFFE TIME	
~ ~	
VERSION DDIFIED ALGORITHM DIST	1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
ENS A S A A A	
FORTE NPS BOEIN	0 0 0 1 1 0 0 0 1 1 1 1 0 0 0 1 1 1 1 0
Σ	
RSION IFIED GORITH DIST	1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
MOD AL	19/1001819010000000000000000000000000000
PLM NPS BOEING TIME	0 0 0 1 1 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1
-	000000000000000000000000000000000000000
ALT	70000000000000000000000000000000000000
TAS	44nnnnaaaawwww444tnnnnaaaawwww444tnnnnaa nnoonnoonnoonnoonnoonnoonnoonnoo
DEG	



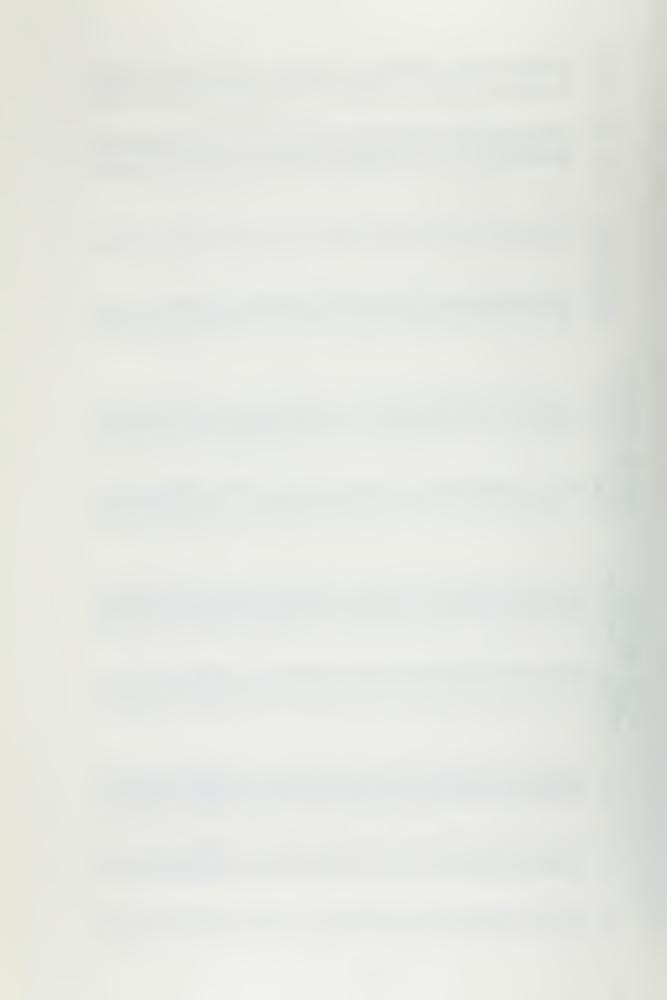
PER CENT ERROR TIME DIST	00000000000000000000000000000000000000
ER ENCES DIST	10010000010100m
DIFF	000000000000000000000000000000000000000
NO VERSION MODIFIED ALGORITHM DIST	1112 1112 1112 1112 64420 64440 6440 64440 64440 64440 64440 64440 64440 64440 64440 64440 64440 6440 6440 6440
FORTRA NPS BOEING TIME	16.07 16.07 16.02 17.02 17.09 17.09 18.02 18.02 19.02 19.03 10.03
ERSION DIFIED LGORITHM DIST	11111121212121212121212121212121212121
PLM NPS MO BOEING A	10000000000000000000000000000000000000
ALT	40000000000000000000000000000000000000
TAS	00000000000000000000000000000000000000
DEG	11111111111111111111111111111111111111



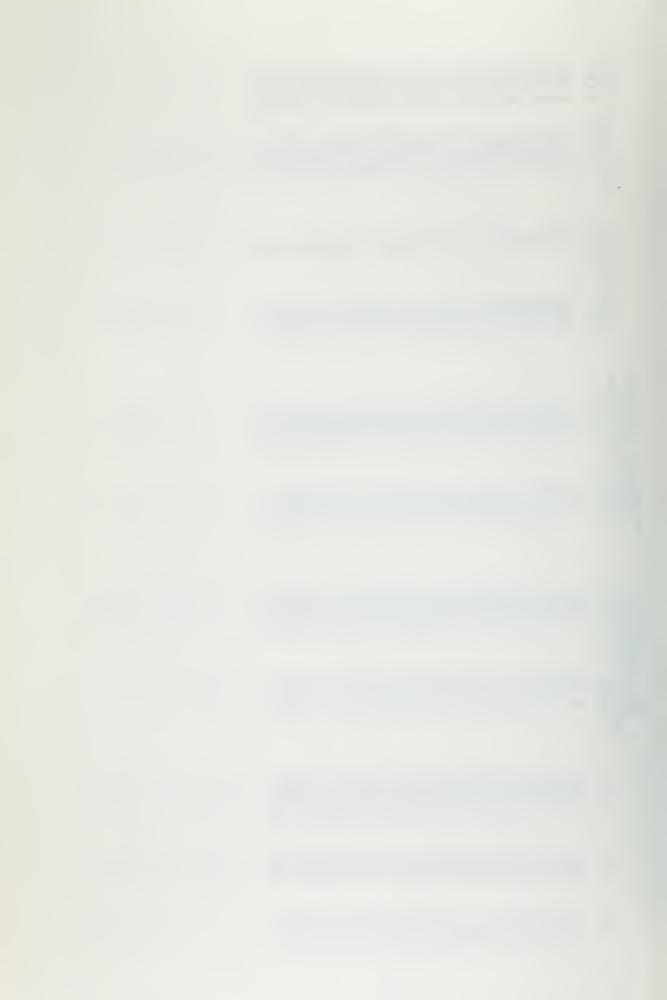
000		T ERROR DIST	
SOS		PER CENT	00000000000000000000000000000000000000
00			
= 70		ER ENCES DIST	
ž.		DIFF	000000000000000000000000000000000000000
		N M	
00	• •	N VERSI MODIFIED ALGORIT DIST	1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
00 0M1 0M2		FORTRA NPS BOEING TIME	
590		Σ	
0.0073290		ERSION DIFIED LGORITHA DIST	1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
11 18	 	204 204	→ できると思いますのこのものものとりととののからものとりととらるとのものもののものとのできるのもののもののもののもののもののもののもののもののものといいます。
FOR ID	MA.	PLM NPS BOEIN TIM	
OEFFICIENTS		ALT	
0 11 11	1 11 11	TAS	www.4444nnnnwwww.444tnnnnwwwww.44 Oonnoonnoonnoonnoonnoonnoo
WEAPON CFORM1	T.Y.P. B.O.T.	DEG	



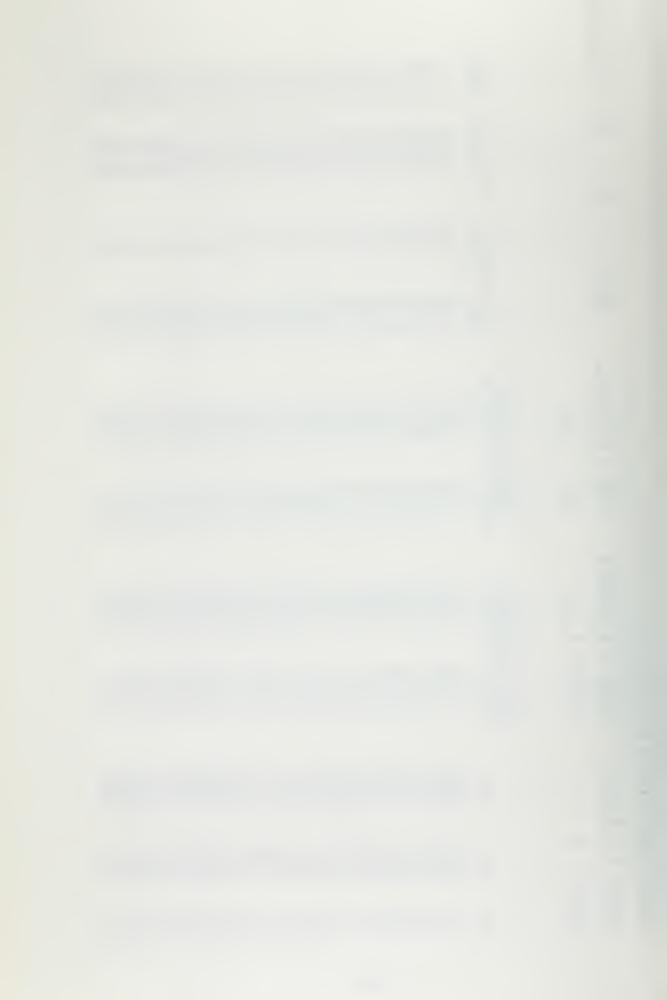
T ERROR DIST	00000	00000	00000	00000	000000	000000000000000000000000000000000000000	00000
PER CENT	-00000	0000-	100000	00000		000000	00000
RENCES		000000		-00000		00000	0-0000
DIFFE	00000	20000	00000	00000	000000	000000	00000
NOTH							
N VERS MODIFIE ALGORI	2463	84 68 68 7 7 8 8 8 8 8 8 8 8	301416	69503 148103	10 10 10 10 10 10 10 10 10 10 10 10 10 1	148365 148365 148383 14183 14183 16063	2550000 2500000
FORTRAN NPS BOEING TIME	466678	25 25 25 25 25 25 25 26 26 27 26 27 27 27 27 27 27 27 27 27 27 27 27 27	32149	₩	40405- 1075414	2 2 2 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4	7450VQ 246474
RSION JFIED GORITHM DIST	252765 202763 202763	00000000000000000000000000000000000000	6499100 914100 9166910	260 260 260 260 260 260 260 260 260 260	0000000 000000000000000000000000000000	1 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4	7000000000000000000000000000000000000
PLM VE NPS MOD BOEING AL TIME	46404	45mm00	35140 44161	₩ ₩₩	していなしら	1 0 0 2 140011221 2 0 0 0 4 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	ひらつら かくををとぬ
ALT	100010000000000000000000000000000000000		421194 600000 600000	000000 000000 000000000000000000000000	44444444 0000040 000000	1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	000000 000000 000000000000000000000000
TAS	N00000	000000 000000	000000	<i>NNOON</i>	0000000		000000
DEG	00000	200000	00000	000000		11111	44444



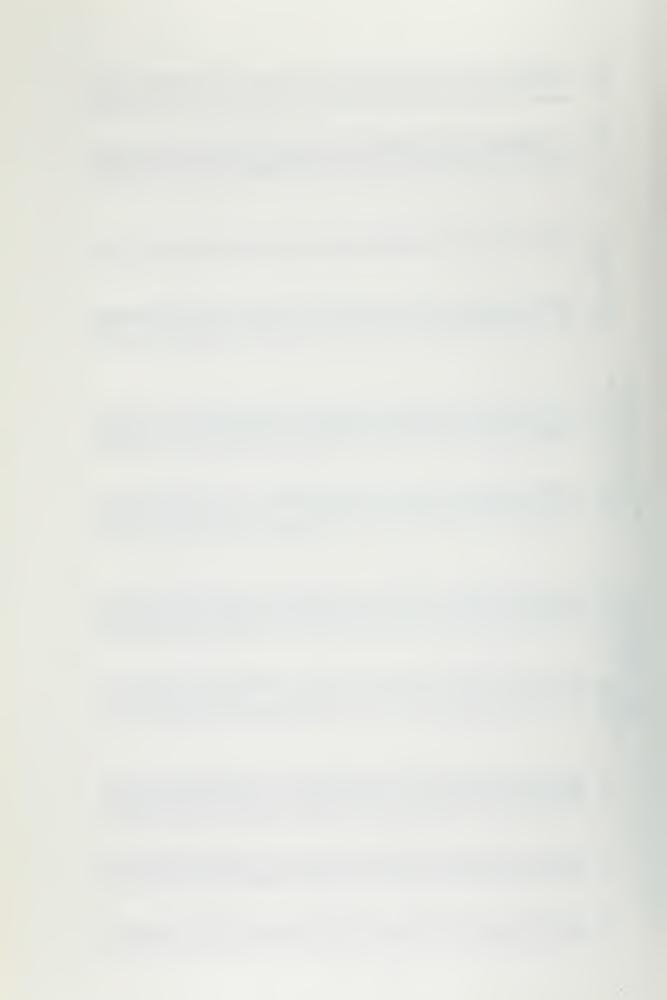
RROR I ST	
NT ER	000000000000000000000000000000000000000
PER CENTINE	000000000000000000000000000000000000000
RENCES DIST	0-00000
DIFFE	000000000000000000000000000000000000000
N W H	
N VERSI MODIFIED ALGORIT	411-7878787878787878787878787878787878787
FORTRAN NPS N BOEING TIME	2 2 2 2 1 1 2 2 2 2 2 8 6 6 6 6 6 6 6 6 6 6 6 6 6 6
5	• • • • • • • • • • • • • • • • • • • •
RS ION IFIED GORITH	
PLM VE NPS MOD BOEING AL	2 2 2 2 1 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2
—	000000000000000000000000000000000000000
AL	4 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
TAS	
DEG	11111111111111111111111111111111111111



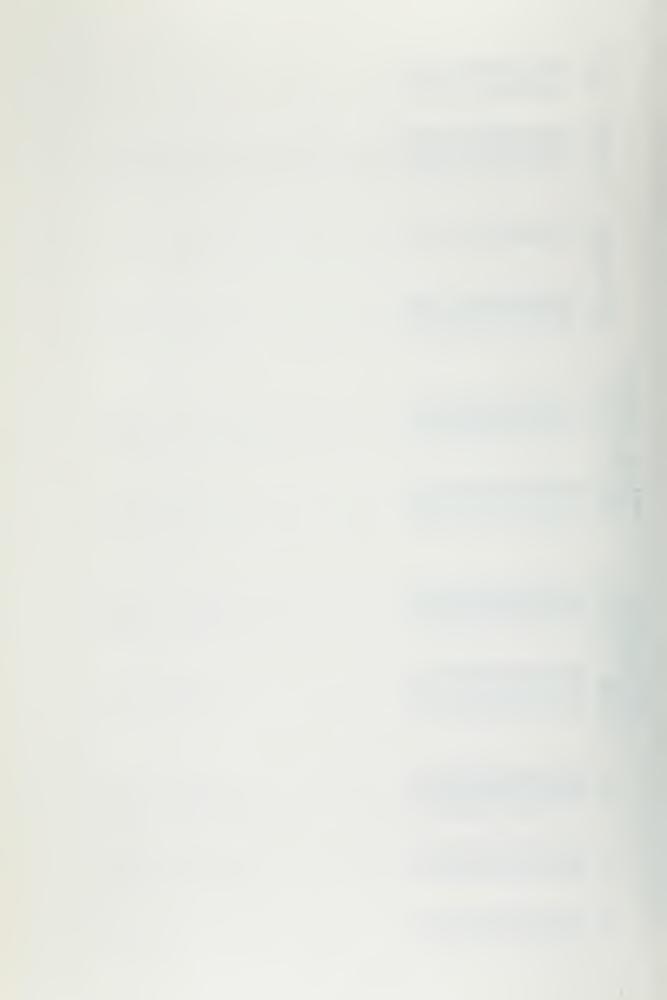
S = 0.6617000 L =0002690	NT ERROR DIST	
00 00	PER CET	
= Zn= -	ERENCES	
O SZ	DIFF	
= 0.0 = 0.3800000 = 2.00	AN VERSION MODIFIED G ALGORITHM DIST	
DM2 DM2 VE DTI	FORTR NPS BOEIN TIME	103000 10300
18 = 0.0073290 = 0.1716599 = 1.	VERSION ODIFIED ALGORITHM DIST	3172121373737373737373737373737373737373
FOR IDNO DKG1 DKG2 IREF DMAX	PLM NPS BOEING TIME	. 1
COEFFICIENTS = 0.0 = 0.0168950 1 2	ALT	
PON RMA1 TH TH	TAS	www.444nnnnwwww.444nnnnwwww.44 OONNOONNOONNOONNOONNOON
WEA CFO CFO I TY I BO	DEG	00000000000000000000000000000000000000



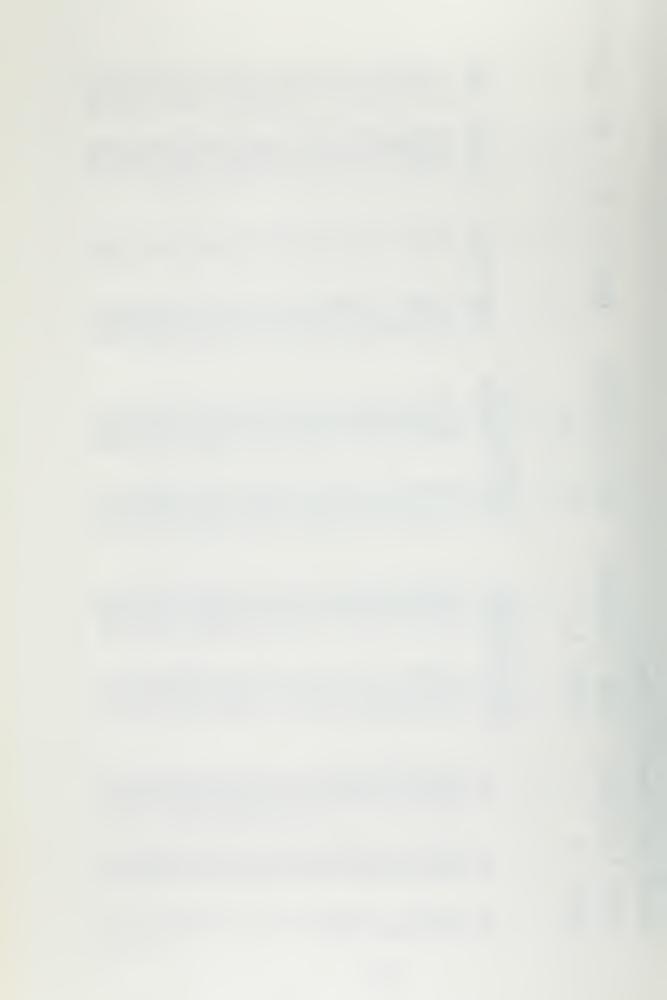
ERROR	
PER CENT	
RENCES DIST	0,0000000000000000000000000000000000000
DIFFE	
VERSION ODIFIED ALGORITHM DIST	00000000000000000000000000000000000000
FORTRAN NPS M BOEING TIME	
ERSION DIFIED LGORITHM DIST	######################################
PLM VI	102800000000000000000000000000000000000
ALT	W = W = W = W V V V V V V V V V V W V W
TAS	44 NN NN MAMA 444 NN
DEG	00000000000000000000000000000000000000

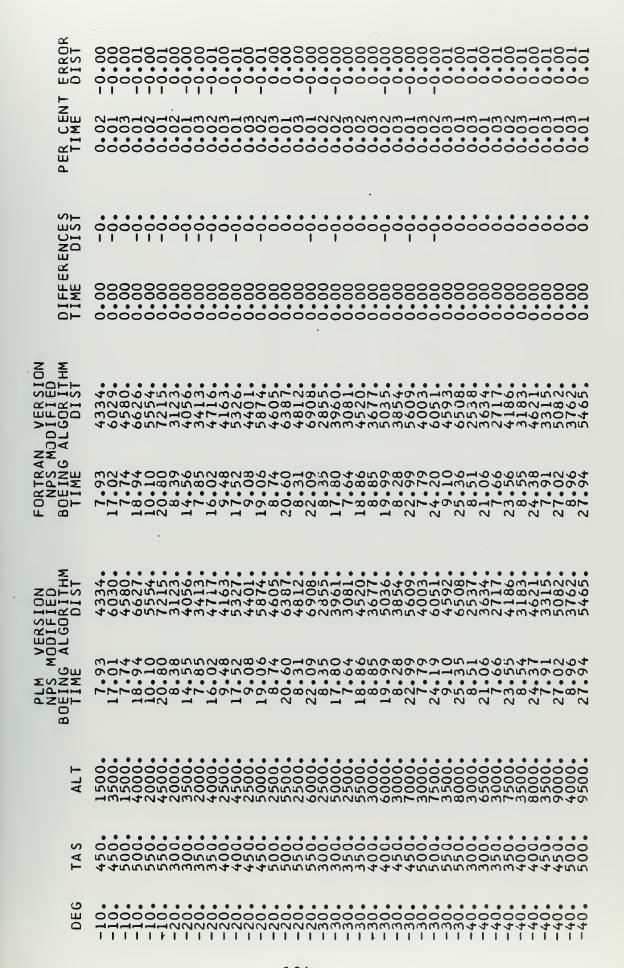


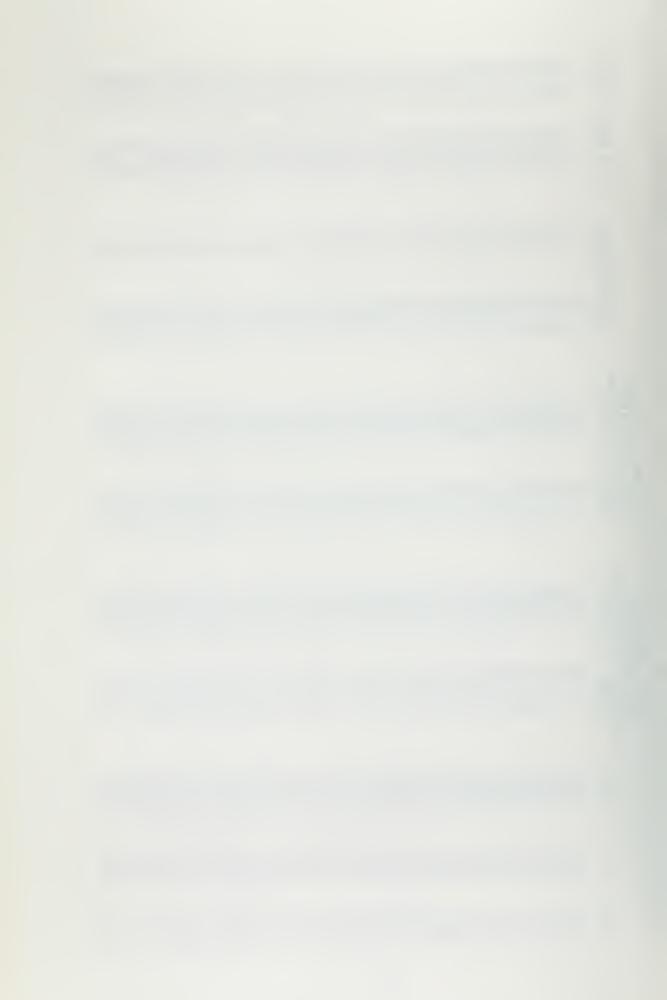
(DIST	0.02	00	0	9	0	9	0	0	0	0	0
i c	PEK CEN-	000										
((DIST	0.1	••	•	•	• •	0	1.	•	٦.	•	!
i i	UITEK TIME	000										
VERSION	LGURI DIS	2646.	9	89	S S	ედ ედ	68	54	41	82	28	0
FORTRAN	I WE	15.00	ω .0.	2.4	9.0	n n n	6.9	5.0	0.2	7.1	1.8	8.3
RSION	GUR I DI	2646.	395 892	489	092	25 28	868	454	041	585	128	200
PL	M S S S S S S S S	15.00	89°1	2.4	0 9	0.00 0.00	6.9	5.9	0.2	7.1	1.8	8.3
	ALT	4500.	88	400	8		5500	8	650	50	8	00
	TAS	55 55 50 50 50	00	S	S	20	S	S	0	0	5	S
	DEG	-45 -45	-609-	9	9	9	9	9	9	9	9	-60



WEAPON CFORM1 CFORM2	COEFF = 2.2 = 0.0	ICIENTS 572994 111360	FOR IDNO DKG1 = DKG2 =	20 = 0,0081750 = 0.1688499	DM1 DM2	0.3200000 0.4100000	VMUZ FN =	"	- av		05 99995
ITYPE :	- 5		IREF =	= 1 5.00	VE = DTI =	2.00		-			
9∃0	TAS	ALT	PLM NPS MC BOEING /	VERSION ODIFIED ALGORITHM DIST	FORTRAN NPS MO BOEING A	VERSION DIFIED LGORITHM DIST	DIFFERE TIME	ENCES DIST	PER CE	NT ERR DIS	TOR.
• • • • • • • • • • • • • • • • • • • •			0.000000000000000000000000000000000000	000412400000000000000000000000000000000	1 1 6 6 6 5 5 2121212019 1 1 2 6 6 6 5 5 212120019 1 1 2 6 6 6 6 6 7 7 7 7 7 7 7 7 7 7 7 7 7 7	0444449000847464746976564047469796064047460817460817460817460817460808071764608081778474608080177847460808177847460808177847460888817784748474847848608474608888178474847484748474847484748474847484		000mmmmmmmmmmmmmmmmmmmmmmmmmmmmmmmmmmm		000000000000000000000000000000000000000	



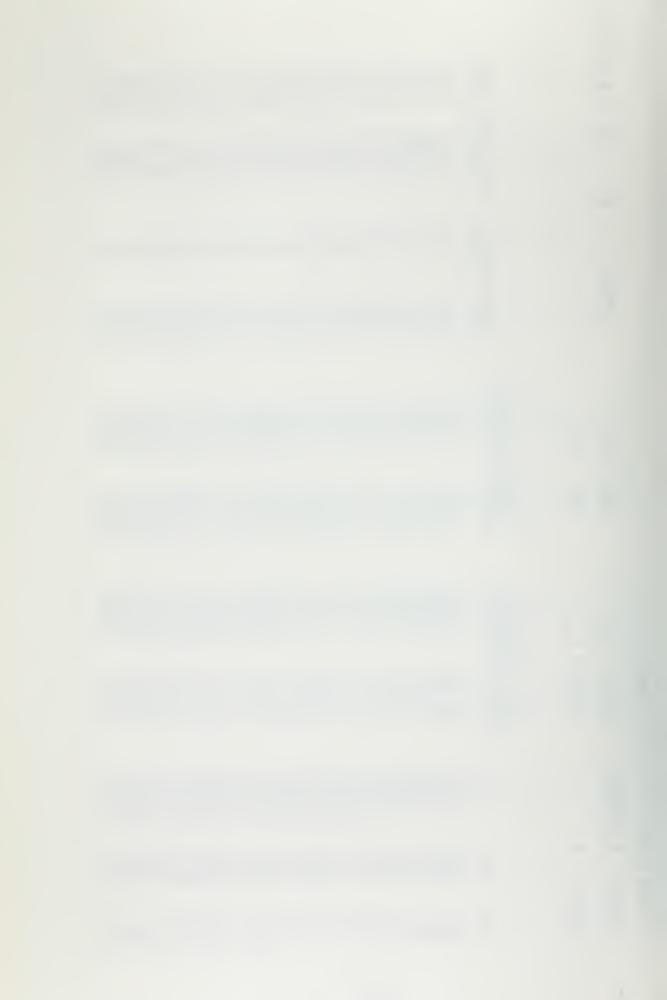




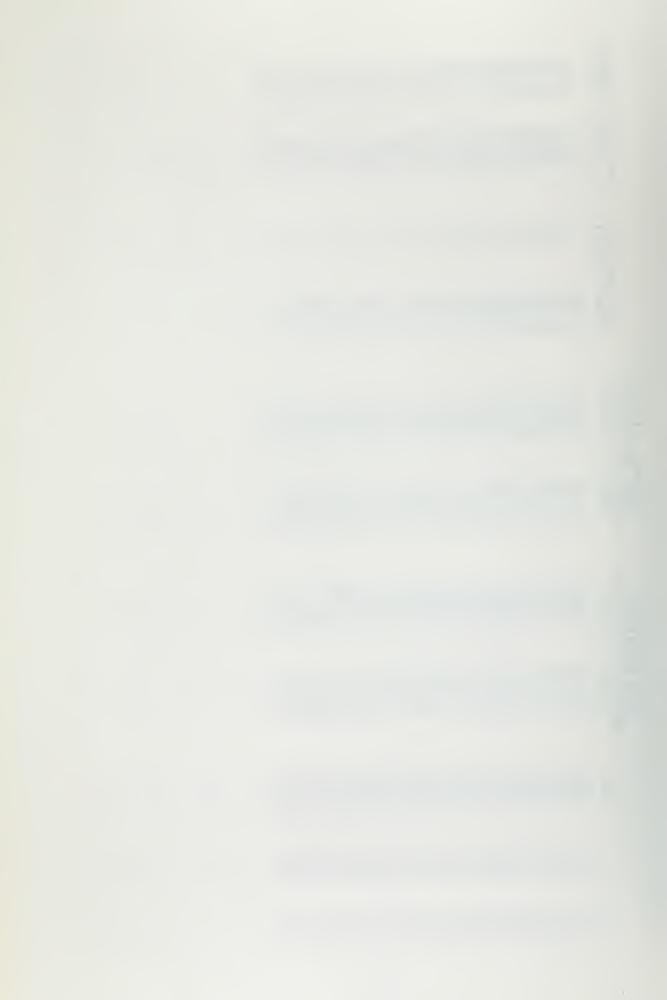
ERROR DIST	
PER CENT TIME	00000000000000000000000000000000000000
ENCES	0-00000
DIFFER TIME	000000000000000000000000000000000000000
MODIFIED ALGORITHM DIST	を は は で で で で で で で で で で で で で で で で で
FORTRAN NPS N BOEING TIME	91414 414 4 8 8 2 2 8 8 4 4 4 4 8 8 9 8 9 8 9 8 9 9 8 9 9 9 9
RSION IFIED GORITHM DIST	4ww2wwww2wwwwwwwwwwwwwwwwwwwwwwwwwwwww
PLM VE NPS MOD BOEING AL TIME	98020204040404040404040404040404040404040
ALT	10000000000000000000000000000000000000
TAS	
DEG	1



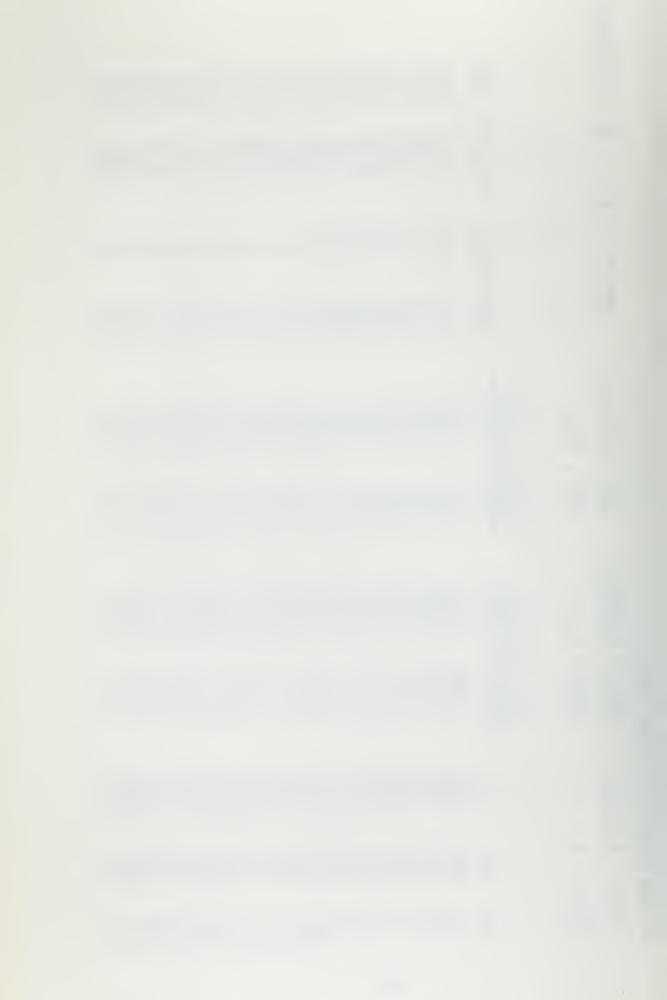
WEAPON	COEFF	ICIENTS	FOR IDNO	21							
CFORM1 CFORM2	= 2.2 = 0.1	178000	DKG1 DKG2	0.00	DM1 DM2	0.0	VMUZ FN =	11	00.0		4.0000000
ITYPE I BOTH	= 2		IREF	= 1 5 00	VE = DTI	= 0.0			·		
DEG	TAS	ALT	PLM NPS BOEING TIME	VERSION ODIFIED ALGORITHM DIST	FORTRAI NPS I BOEING TIME	MODIFIED ALGORITHM DIST	DIFFE	RENCES	PER CEI	N D	RROR I ST
000000000000000000000000000000000000000	444NNNN4444NNNN4444NNNN44444NN OONNOONNO	м м м м плититимими м м м м м м м м м м м м м м м	21222222222222222222222222222222222222	70000000000000000000000000000000000000	21222222222222222222222222222222222222	 πυνου ο ο ο ο ο νου νου νου ο ο ο ο ο ο ο		0-0-0-000000000000000000000000000000000	000000000000000000000000000000000000000	000000000000000000000000000000000000000	000000000000000000000000000000000000000



T ERROR DIST	000000000000000000000000000000000000000
P ER CEN	
RENCES	
DIFFE	
MODIFIED ALGORITHM DIST	4 W 4 W 4 W 4 W 4 W 4 W 4 W 4 W 4 W 4 W
FORTRA NPS BOEING TIME	### ##################################
IFIED GORITHM DIST	4W4W4WWW4444WAWAAWAAWAAAAAAAAAAAAAAAAA
PLM VE NPS MOD BOEING AL TIME	######################################
ALT	имимимимимимимимимимимимимимимимимимим
TAS	WW4444WWWW4444WWWW4444WWWW WWOOWWOOWWOOW
DEG	



WEAPON C	COEFFICIENTS		FOR IDNO	22						
CFORM1 = CFORM2 =	0.0	0625	DKG1 = DKG2 =	= 0.0097670 = 0.2328700	DM1 = DM2 =	0.0	VMUZ FN =	11	0. 0. SL	= 0.6790000 =0003030
ITYPE = IBOTH =	-2		IREF = DMAX =	= 1 5 • 00	VE = OTI =	0.0				
DEG TA	A Si	ار T	PLM V NPS MO BOEING A	/ERSION JDIFIED ALGORITHM DIST	FORTRAN NPS M(BOEING /	VERSION ODIFIED ALGORITHM DIST	DIFFER	ENCES DIST	PER CE	NT ERROR DIST
		000000000000000000000000000000000000000	1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	21212121212121212121212121212121212121	4849404011111111111111111111111111111111	21212121212121212121212121212121212121	000000000000000000000000000000000000000	000000000000000000000000000000000000000	000000000000000000000000000000000000000	



NT ERROR DIST	
PER CEN	000000000000000000000000000000000000000
RENCES	
DIFFE	000000000000000000000000000000000000000
MODIFIED ALGORITHM DIST	2222222222222222222222222222222222222
FORTRA NPS BOEING TIME	
ERSION DIFIED LGORITHM DIST	22222222222222222222222222222222222222
PLM NPS MO BOEING A	
ALT	
TAS	44NNNNWWWWWW4444NNNNWWWWWW4444NN NNOONNOONNOONNOONNOONNOONNOONNOON
DEG	



ERROR DIST	000000000000000000000000000000000000000
PER CENT	000000000000000000000000000000000000000
ENCES DIST	
DIFFER	000000000000000000000000000000000000000
VERSION ODIFIED ALGORITHM DIST	2222 2224 2495 1125 2523 1126 1126 1136 1136 1136 1136 1136 11
FORTRAN NPS MO BOEING	121 121 123 124 125 126 136 136 136 136 136 136 136 136 136 13
IFIED GORITHM DIST	22278 24828 12557 13496 15536 17752 1775 1986 1775 2054
PLM VEI NPS MOD BOEING AL	22
ALT	45000 60000 1000000 105000 115000 12000 12000
TAS	00000000000000000000000000000000000000
DEG	11111111111111111111111111111111111111

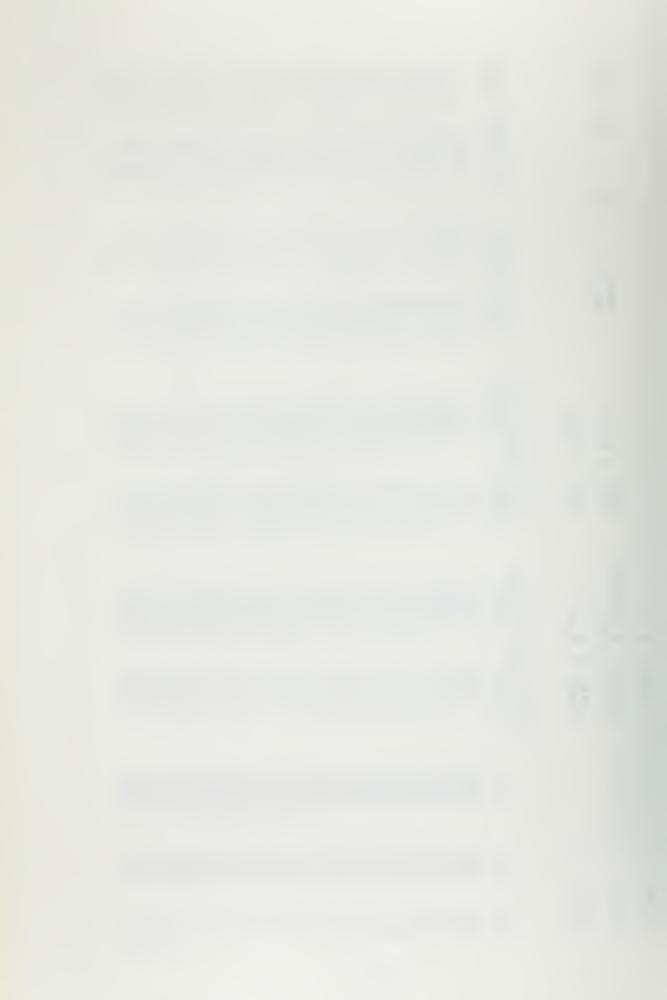


APPENDIX B

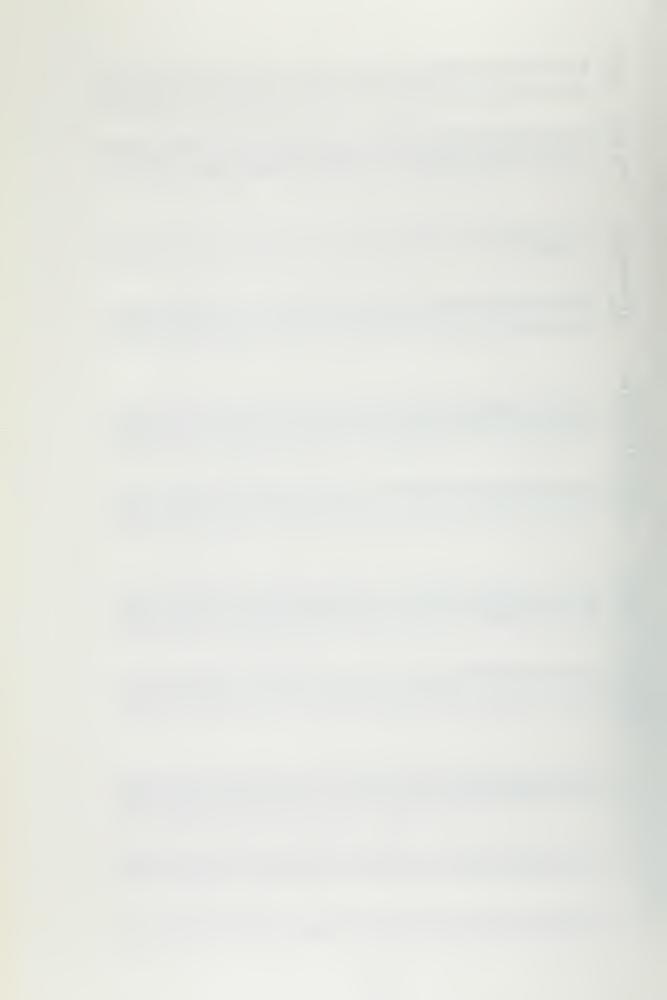
This appendix compares the NAVAIR 01-1C-1T-1 Ballistics
Tables with the results of the FORTRAN version of the
ballistics algorithm. The difference in down range travel
and time of fall is presented.



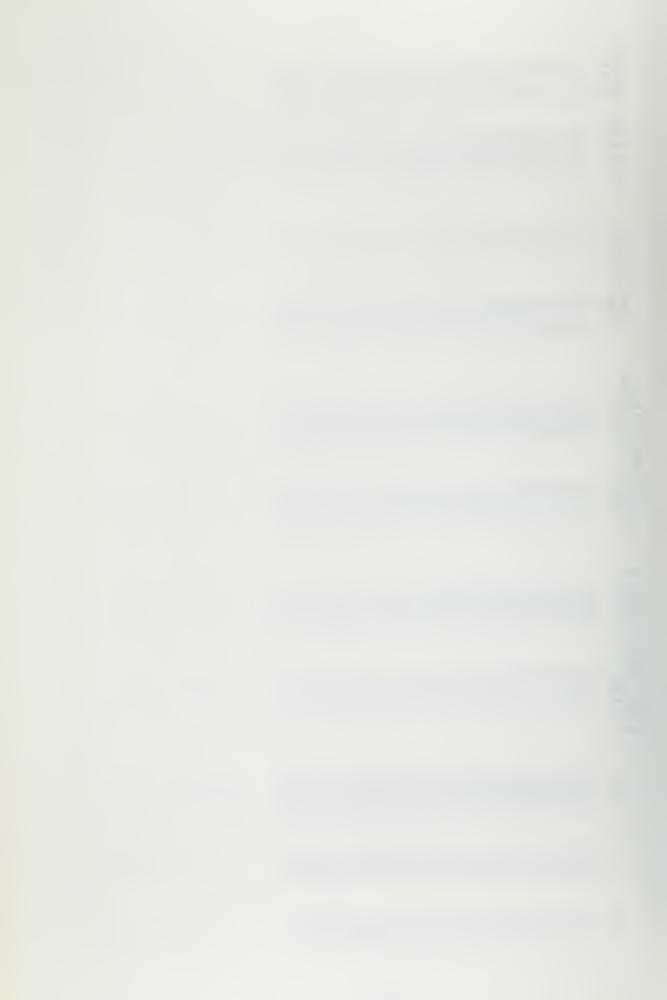
0 • 0 = 0 • 0 0 • 0 = 7 S	PER TERM TERM TERM TERM TERM TERM TERM TE
VMUZ = FN =	DIFF ENCES TO 000 000 000 000 000 000 000 000 000 0
DM1 = 0.0 DM2 = 0.0 VE = 0.0 DTI = 2.00	NPS MODIFIED BUEING ALGORITHM 8 94
DKG1 = 0.0027540 DKG2 = 0.0 IREF = 2.00	JAVAIR 01-1C-1T-1 11ME 01-1C-1T-1 8.96 17590. 17.666 7590. 17.666 7590. 17.666 17590. 18.24 1724. 18.24 1724. 19.94 1374. 19.94 1724. 19.94 1724. 33.30 13157. 19.94 1744. 33.30 15780. 33.94 21390. 33.94 22838. 12.81 1966. 33.94 21390. 33.94 21390. 34.18 22838. 14.759 1716. 15.80 1716. 16.28 1716. 16.28 1987.
0.0039235 0.0039235	A W W W W W W W W W W W W W W W W W W W
WEAPON COE CFORM1 = 0 CFORM2 = 0 ITYPE = -1 IBOTH = 1	



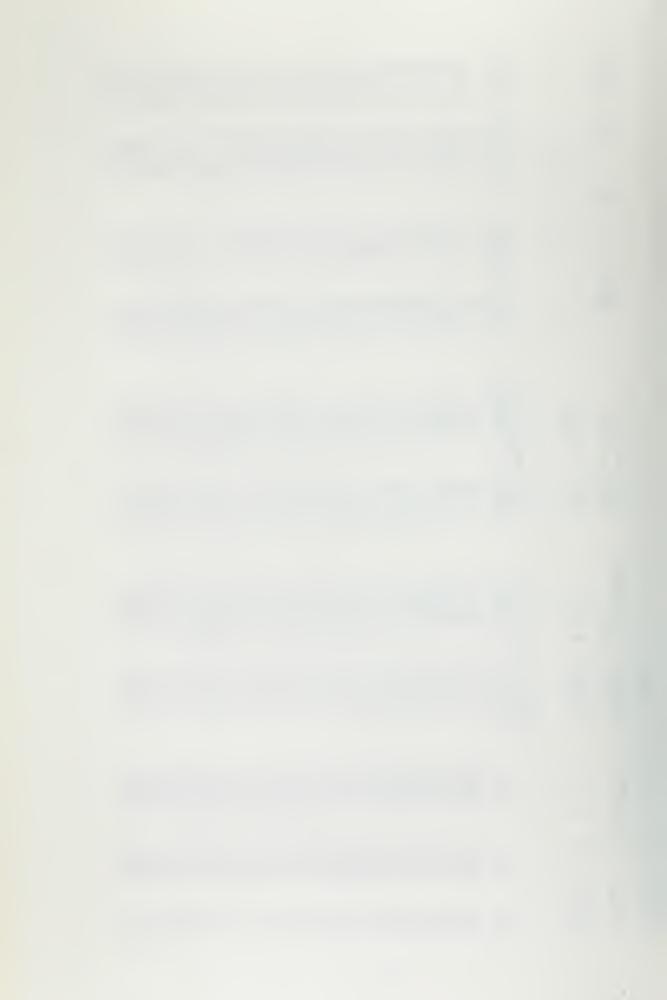
PER CENT ERROR TIME DIST	
ER ENCES DIST	
DIFF	000000000000000000000000000000000000000
MODIFIED ALGORITHM DIST	1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
NPS BOEING TIME	2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
CS TABLES DIST	1 1 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
NAVAIR O BALLISTI TIME	2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
ALT	1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
TAS	44NNNNWWWWWW4444NNNNWWWWW4444NN NNOONNOONNOONNOONNOONNOONNOONNOO
DEG	



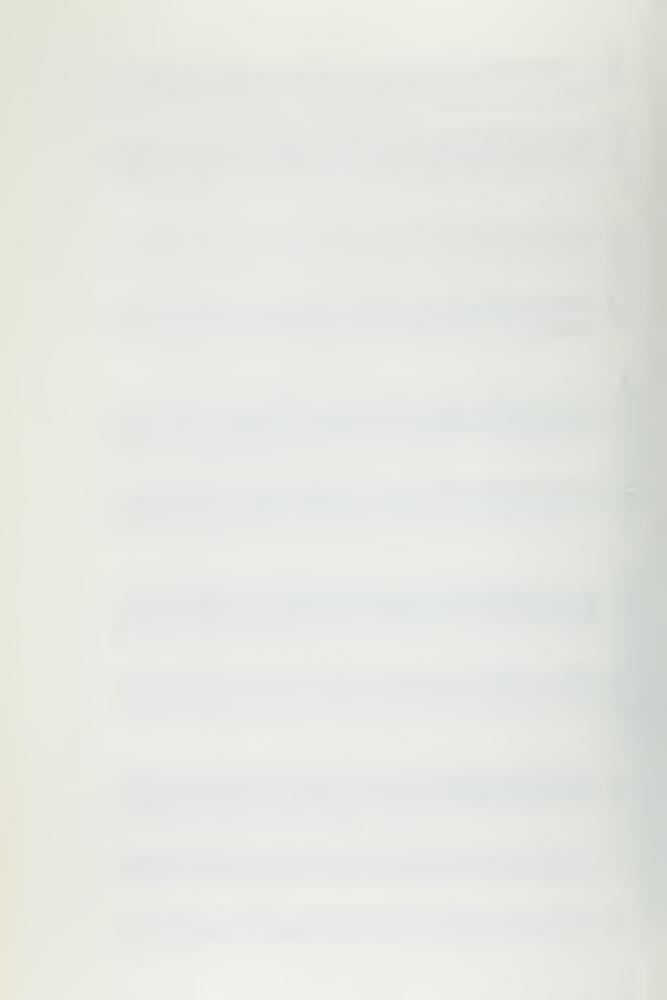
CENT ERROR	02 -0		0.00	0.0- 50	08 -0-1	11 -0-1	03 -0•1 06 -0•1	16 0.0	14 -0-0	20 -0.0	19 -0-0	0.0- 50	15 -0.0	21 -0-1	100-	0.0
S PER TI	000		000	0		00		0		0	00	0				
FER ENCE E DIS	Omr	-	1 2	i I	0") ·	-1 0 1	· CO •	! 	1	 7-	Í	31	2	00	o LO
DIF	000	000			00		00	0	00	0.00		0	00	0		
MODIFIED ALGORITHM DIST	4048.	75.	+∞r -∞-	52 52	96	400	8 2 2 3	87	2 2 3 3	89	3/	54	66 84	15	017 م	30
NPS BOEING TIME	901	9.	ס אוטַו	2.0 1.6	ινς Φα	6.2	6.5	91	1.92	90	70.	9.6	7.6	8	6. 4	7.3
Ol-1C-1T-1 ICS TABLES DIST	4055.	757	48°	52	96	4	52 86 86	87	3 2 3 3	89	76	54.	66 85	15	01 م	200
NAVAIR BALLISTI TIME	901	-9-		1.6	50.0	6.2	0.1	91	26	90	7.6	9.6	7.6	8	ς α • α	7.4
ALT	4000 5000	200	2000	2000 2000 2000	3500	4000	5000 4500	5000	4000 5000	4000	5000 5000 5000	5000	5500 5000	6500	5000	5000
TAS	000 000 000	004	200	00	500	00	0r 00	500	00	500	20 20 20	000	S S S	000	000) ()
DEG	400		44 UN	44 50	4. 7.	44	450	45	60 60 60	609	909	9	60	60	909	909



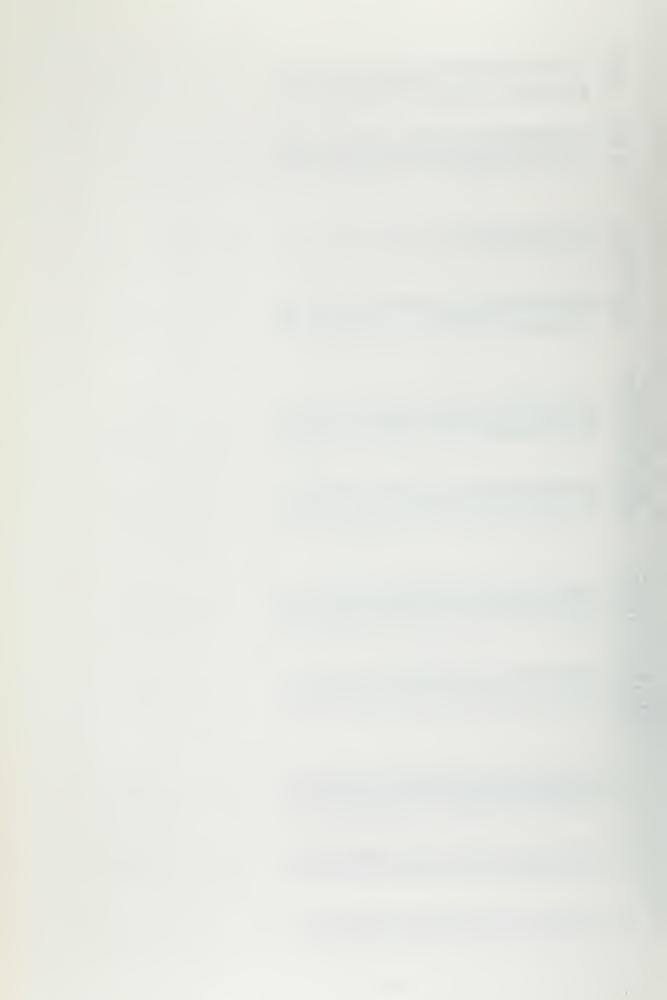
	000		T ERROR DIST	00000000000000000000000000000000000000
	DS SL		PER CEN TIME	00000000000000000000000000000000000000
	00			
	= = =		ERENCES DIST	
	NA PNA		DIFFE	000000000000000000000000000000000000000
			Ψ	
	00	0.0	OD IFIED ALGORIT DIST	2 2 1 1 1 1 1 06 8 5 7 4 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
	DM1 = DM2 =	. VE =	NPS M BOEING TIME	1 11 11 1
	248		ES S	
2	0.0063648	3.00	S TABLI DIST	2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
	11 11	11 11	R 01-	トレー の に の に の に の に の に の に の に の に の に の
FOR IDNO	DKG1 DKG2	IREF	NAVAI B BALLI S TIME	
-ICIENTS	0039077		ALT	
N COEFFIC	1 = 0. 2 = 0.	1	TAS	
WEAPON	CFORM	I TYPE I BOTH	DEG	



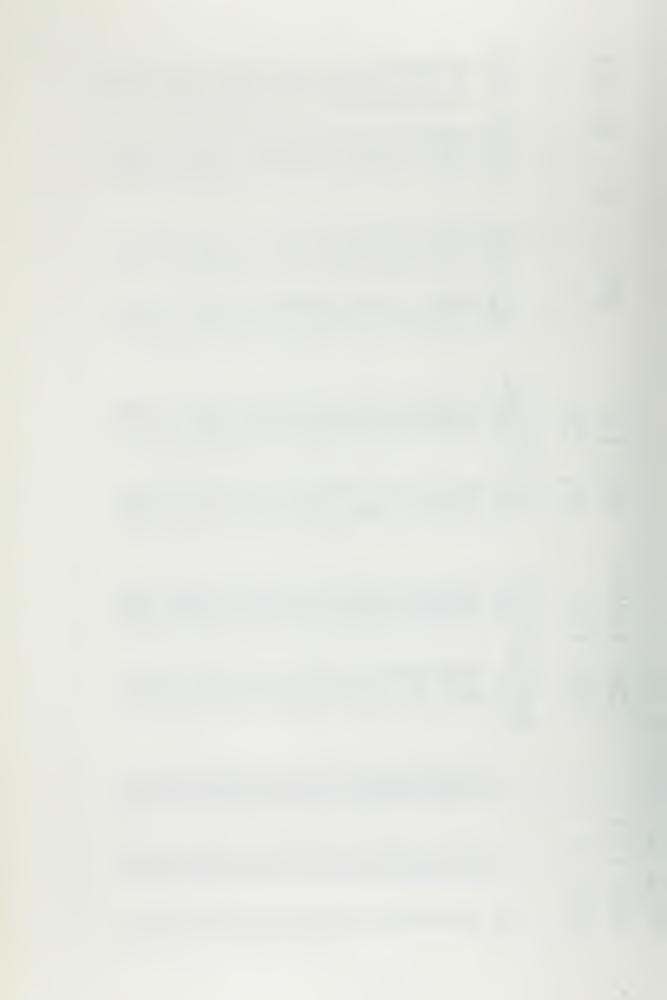
PER CENT ERROR TIME DIST	00000000000000000000000000000000000000
RENCES	
DIFFE	000000000000000000000000000000000000000
MODIFIED ALGORITHM DIST	1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
NPS BOEING TIME	2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
1-1C-1T-1 CS TABLES DIST	1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
NAVAIR O BALLISTI	1 1 2 1 1 1 1 2 2 1 1 2 2 2 2 2 2 2 2 2
ALT	1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
TAS	44NNNNWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWW
DEG	11111111111111111111111111111111111111



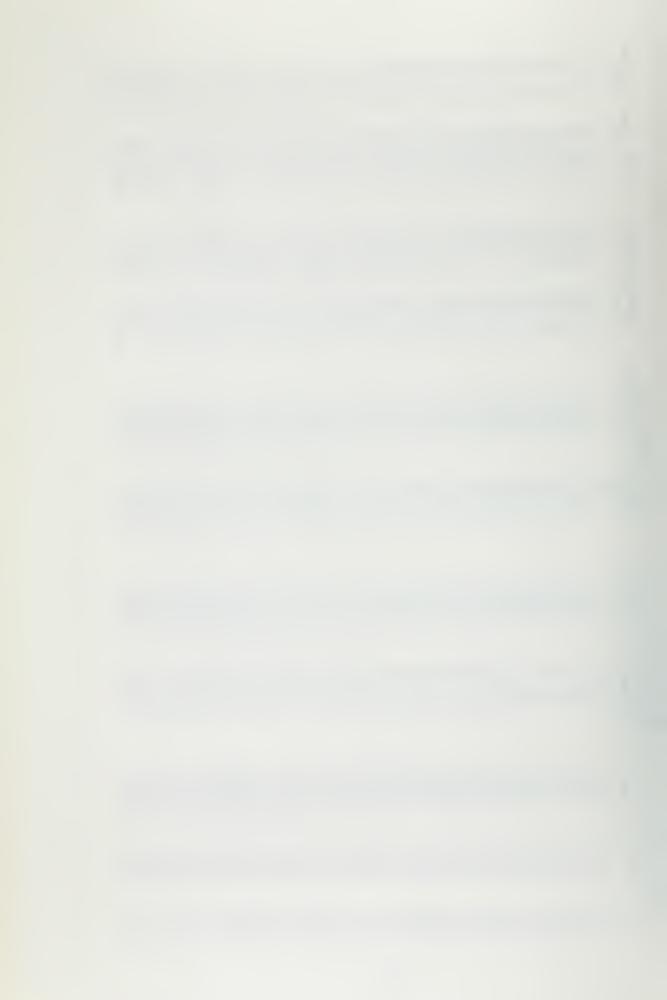
					٠					
IT ERROR DIST	0000	0011000	000		-00	000	000	000	000	0
P ER CEN TIME	7000	70m- 0000 1 1	000		-00	000	000	0	000	0
ERENCES DIST		 			970		4-10		204	
DIFF	000	0000			000		000	000	000	
MODIFIED ALGORITHM DIST	993	7022• 2411• 7746• 2509	2000	307	424	882	360	402	13	23
BOEING	4-16	24 • 21 6 • 18 23 • 15 5 • 6	520	-00 -00 -00	6.6 0.1 7	2.40	1.2	7.6	ໝໝ ເບ _{ົາເບ} ິດ	7.00
1-1C-1T-1 CS TABLES DIST	900	7035 2414 7753	500	307	84 74 10	66 89	36	407	14 03 42	23
NAVAIR O BALLISTI TIME	4-18	24•21 6•18 23•16 5-16	200	001	900	6.2	7-5	970	888 5044	200
ALT	2500	150000 150000 150000	3000	500 500 500	4500 5000 5000	5000 5000 4000	5000	5000 5000 5000	5000	500
TAS	000	00000000000000000000000000000000000000	000	000	200	200	000	2000	000	200
DEG	444 000		よなら	444 2121	444	909	600	900	60	909



	DS = 0.0 SL = 0.0		PER CENT ERROR TIME DIST	2000 20 20 20 20 20 20 20 20 20 20 20 20
	00			
	= Zn		ERENCES DIST	1 1 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
	>IT		OI FF	00000000000000000000000000000000000000
	00000	= 0.0 = 1.00	MODIFIED G ALGORITHM DIST	1038934409494899594000000000000000000000000000
	DM1 DM2	VE DTI	NPS BOEIN TIME	100 100 100 100 100 100 100 100 100 100
9	0.0212660	2.00	1-1C-1T-1 CS TABLES DIST	2024 10 10 10 10 10 10 10 1
FOR IDNO	0KG1 = 0KG2 =	IREF = DMAX =	NAVAIR O BALLISTI	1006. 1006.
FICIENTS	00		ALT	1
PON COEF	$M_2 = 0.0$		TAS	wwww4444nnnnwwww4444nnnnwwww44 OONNOONNOONNOONNOONNOO OOOOOOOOOOOO
WEAPO	CFORN	IBOTH	DEG	00000000000000000000000000000000000000



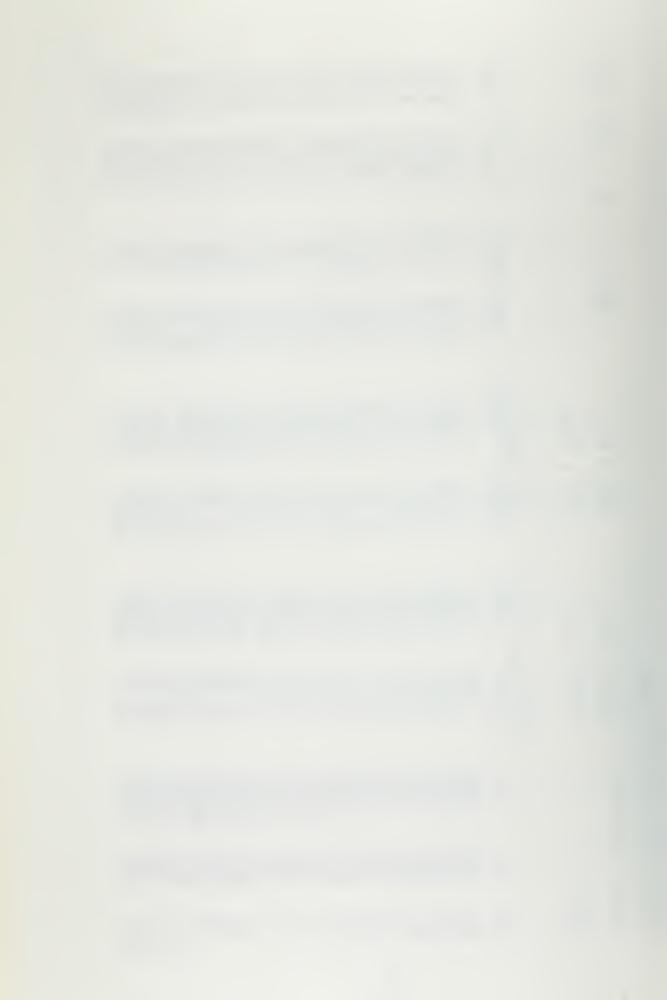
T ERROR DIST	2 2 2 2 8 8 8 2 2 2 8 8 8 8 2 2 2 8 8 8 8 7 2 2 8 8 8 8
PER CENTINE	
RENCES	2 2 1 1 2 2 1 1 1 2 2 1 1 2 2 2 1 1 2
DIFFE	
MODIFIED ALGORITHM DIST	$\begin{array}{c} W L w a w Q u d v d d v d d d v d d d d d d d d$
NPS N BOEING TIME	1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
S TABLES DIST	るというというというというというというというというとしてものもしているというというというというというというというというというというというというというと
NAVAIR 01- BALLISTICS TIME	2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
ALT	1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
TAS	44NNNNWWWWW4444NNNNWWWWWW4444NNNNWWWWWWW
DEG	

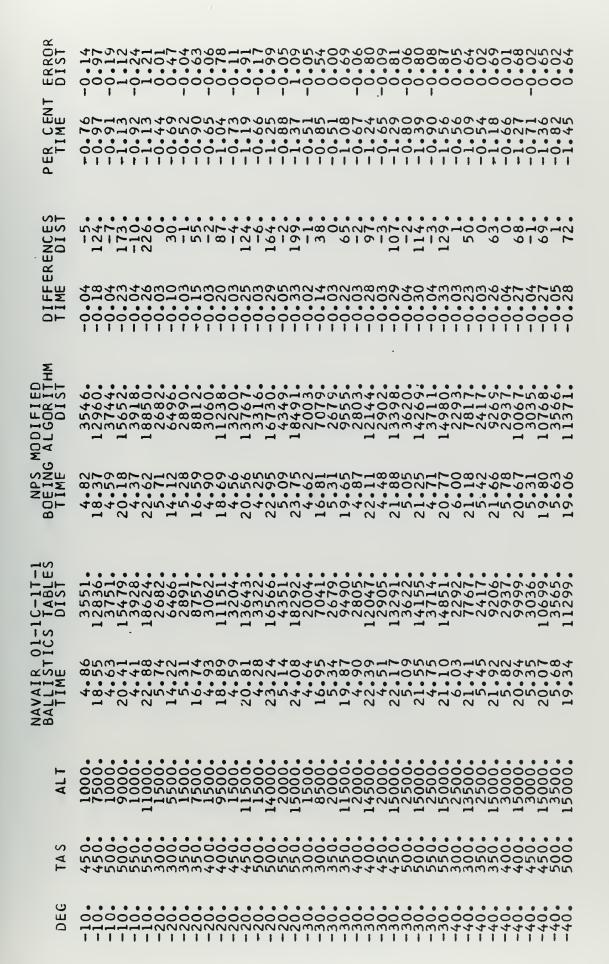


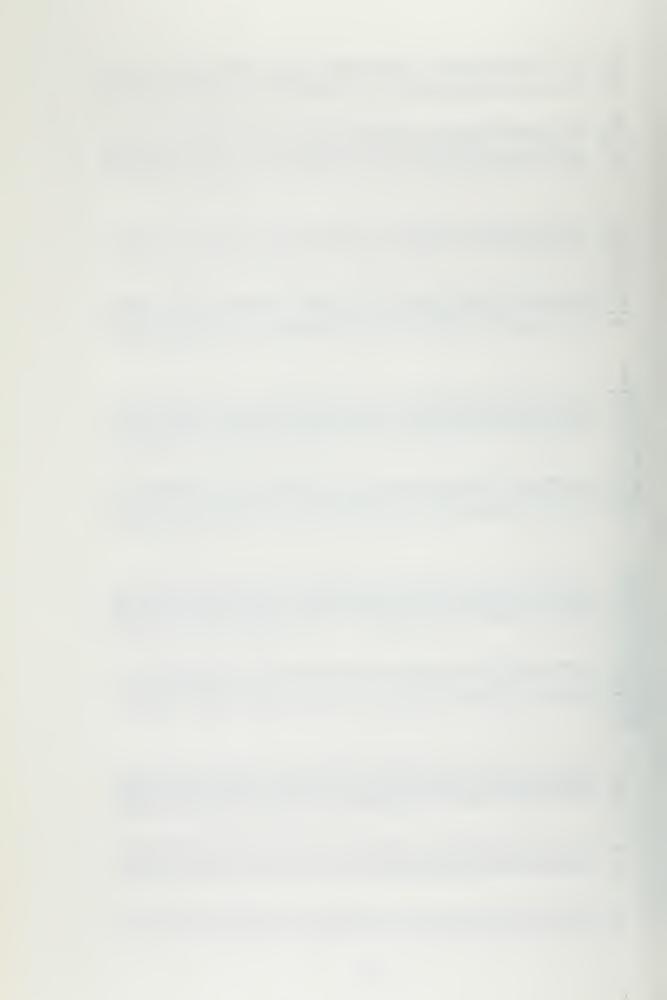
T ERROR DIST	24 2 3 3 4 4 8 4 4 8 9 9 9 9 9 9 9 9 9 9 9 9 9 9
P ER CEN	######################################
RENCES DIST	2316 2316 1212328 122236 123236 1334236 1334326 1334326 1334326 1334326
DIFFE	-12.44 -11.054 -10.33 -10.33 -10.33 -10.33 -10.33 -10.33 -10.33 -11.00 -11.00 -14.00 -
MODIFIED ALGORITHM DIST	6 W W W W W W W W W W W W W W W W W W W
NPS NOEING TIME	20.32 23.24 23.24 22.03 22.00 20.86 19.95 19.81 17.95 17.95 17.95
1-1C-1T-1 CS TABLES DIST	4342429292942 4342429292929 4359952942 4359952942 43599524
NAVAIR OF BALLISTIC	3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3
ALT	15.500000000000000000000000000000000000
TAS	NWWWWW4444WWWW NWOOWWOOWW 00000000000000
DEG	00000000000000000000000000000000000000



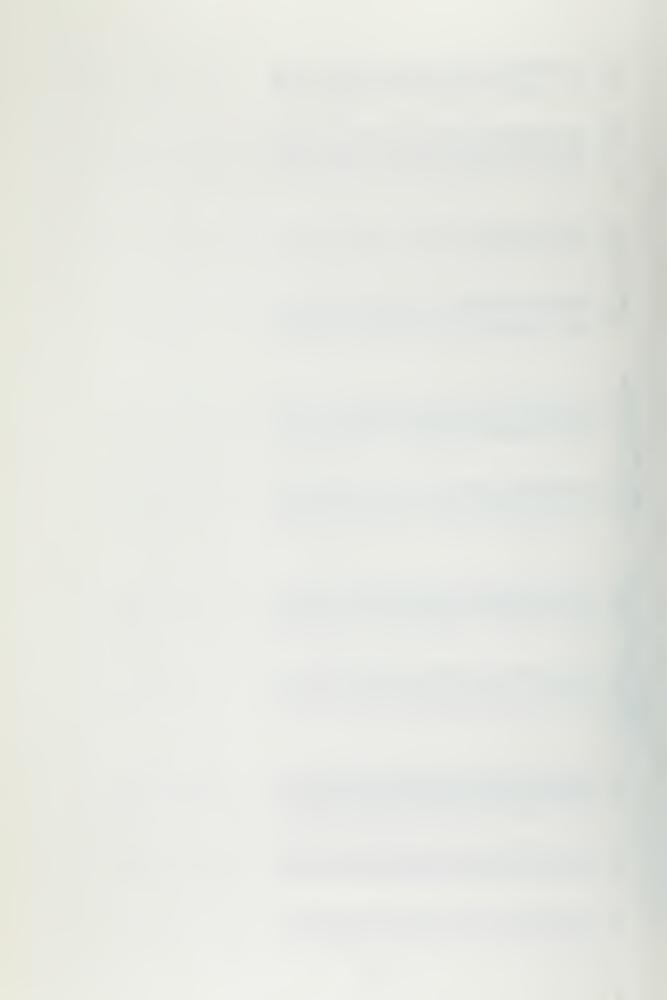
d	00		T ERROR DIST	00000000000000000000000000000000000000
C	SL		PER CENT	00000000000000000000000000000000000000
	0			
= 12	J		ERENCES DIST	1 2 2 2 1 3 2 2 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4
	T.		DIFFE	11111111111111111111111111111111111111
			Ξ	
c	00	00.0	MODIFIED ALGORITE DIST	2 2 2 1 1 1 1 1 2 2 2 2 2 2 2 2 2 2 2 2
- E	DM2	VE = DTI	NPS P BOEING TIME	1 11 11 11 11 11 11 11 11 11 11 11 11 1
			ES	••••
•	00	5.00	-1C-1T S TABL DIST	1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
0 "	11	11 11	1 01 T I C	1108077017080171677777777777
01	DK62	IREF	NAVAIR BALLIS TIME	
COEFFICIENTS	6601		ALT	11000000000000000000000000000000000000
	M2 = 0.0	11 " "	TAS	www.444wwwwww.444wwwwwww.44 000000000000
EA	CFOR	I BOT	DEG	000000000000000000000000000000000000000





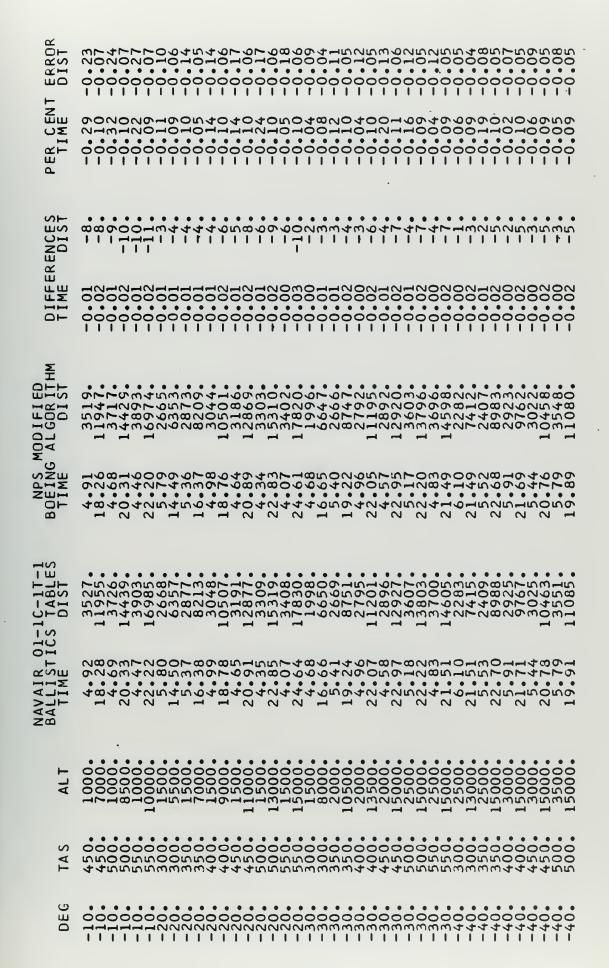


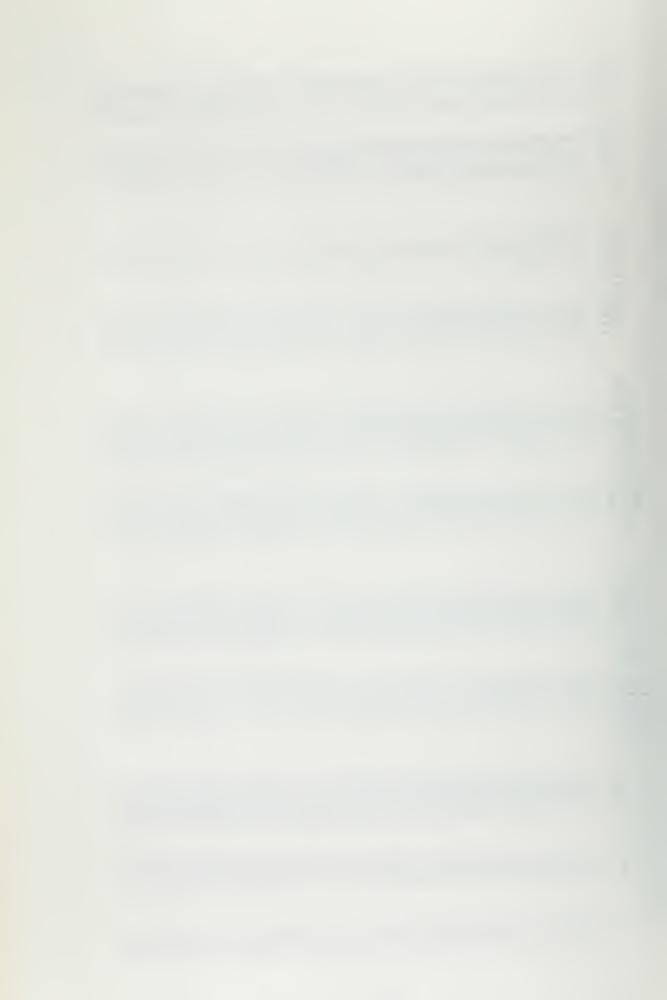
	T ERROR DIST	00000000000000000000000000000000000000	-15
	PER CENT		1.1
	RENCES DIST	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	
	DIFFE	00000000000000000000000000000000000000	0.0
	ALGORITHM DIST	1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	240
	NPS M BOEING TIME	1	ວເກ ອານ
	1-1C-1T-1 CS TABLES DIST	114-11-11-11-11-11-11-11-11-11-11-11-11-	10 10 10 10
	NAVAIR BALLISTI TIME		S S S S S S S S S S S S S S S S S S S
	ALT	4 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	5000
	TAS	ипишии4444ипишиши4444ипи писописописописописоп 000000000000000000000000000000000000	200
	DEG	1	9



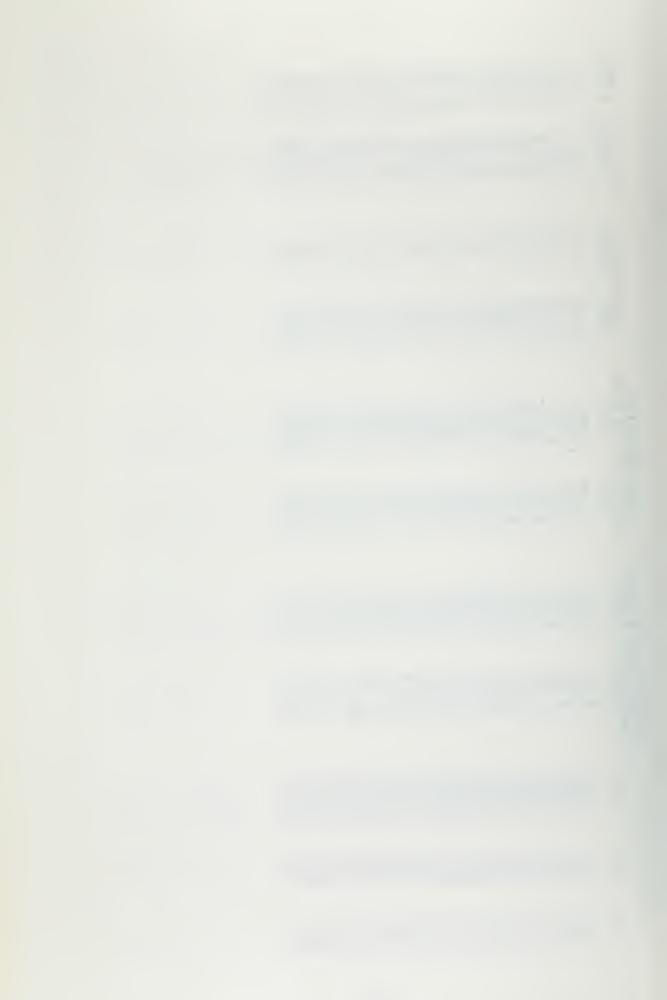
00000	00000000000000000000000000000000000000
SL	T
00	
= 70	M
D WZ >U	ΠΗ 000000000000000000000000000000000000
•	Σ I
000000000000000000000000000000000000000	AD A C G G G G G G G G G G G G G G G G G G
DM1 = VE = DTI = DTI = DTI	0 1 11111111 8 8 8 8 8 1 1 1 1 1 1 1 1 1
8 0.0097670 0.0 3.00	10-11-11 10-11-
FOR IDNO DKG1 = DKG2 = IREF = DMAX =	BNA
COEFFICIENTS = 0.0 = 0.0 -1	ALT NO 00000000000000000000000000000000000
11 11	- wwww4444nnnnwwww4444nnnnwwww444 4 OONNOONNOONNOONNOONNOO N OOOOOOOOOOOOO
WEAPON CFORM1 CFORM2 ITYPE	© 000000000000000000000000000000000000





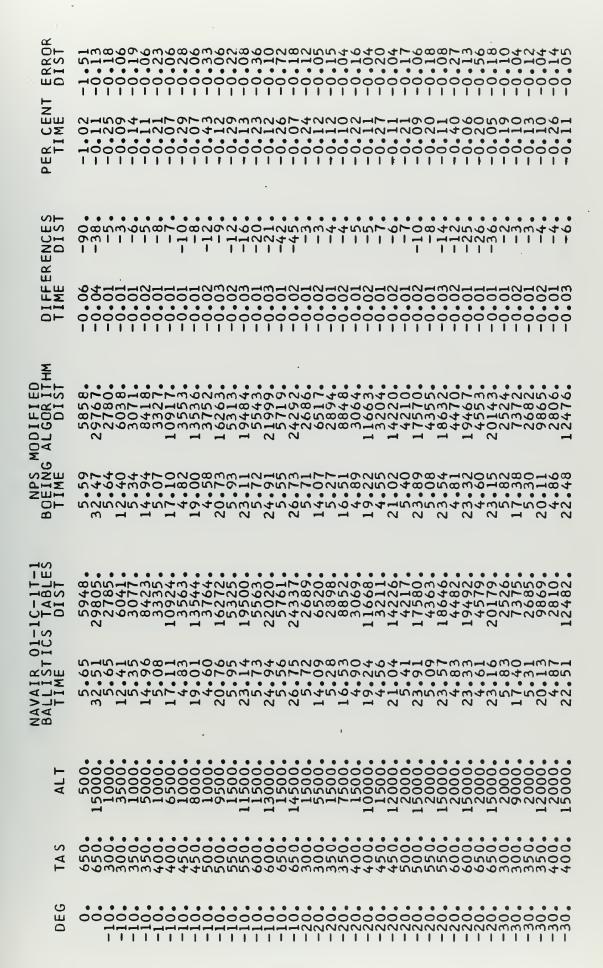


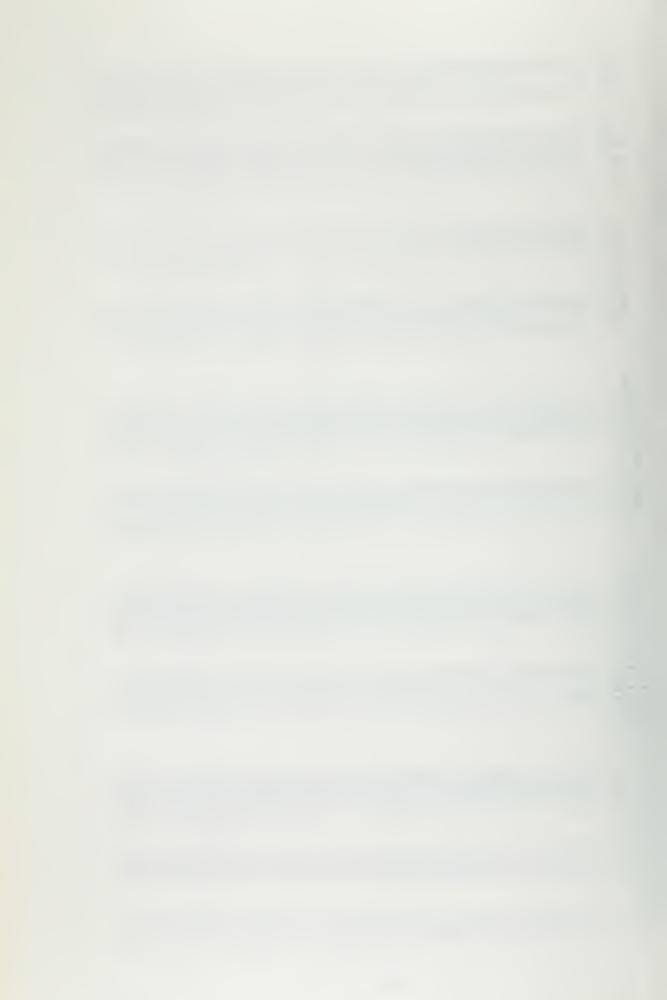
IT ERROR DIST	0.00	000	000	00	00	00	00	000			00	000
PER CEN	-0.01	000	000	00	00	00.0	0.1	000	000		00	000
RENCES DIST	1			S		5						
OIFFE TIME	11000	000	00	00	00	00	00	000		0	00	00
MODIFIED ALGORITHM DIST	4080- 11637- 1983-	140°C	960	281 426	8 8 8	261 807	804 900	2 2 8	637	96	1 2 5	500
NPS P BOEING TIME	19.08	Oor	10m	90.9	8 9 9 9	8.1 7.4	1.3	0.0	8	7.8	α ν Ο α	89
STABLES DIST	4083. 11642. 1984.	140r 50w 50w	100	4 2 8 4 7 8	980 388 9	26 80 80	90	35	60,4	6	10	140
NAVAIR 01 BALLISTIC TIME	19.09	100 c	, 000, 000,	9.8 6.0	6.3	8.17.4	1.3	0.1	8.9	7.8	00,0	000
ALT	15000.		5000	5000 4000	5000 4500	5000 4000	5000	5000	5000	5000	6500	5000
TAS	0000 0000 0000	0000	200	500	900	000	900	200	000	20	00	200
DEG	1111 4444 0000	744	44	452	450	42 60 60	909	909	909	909	909	999



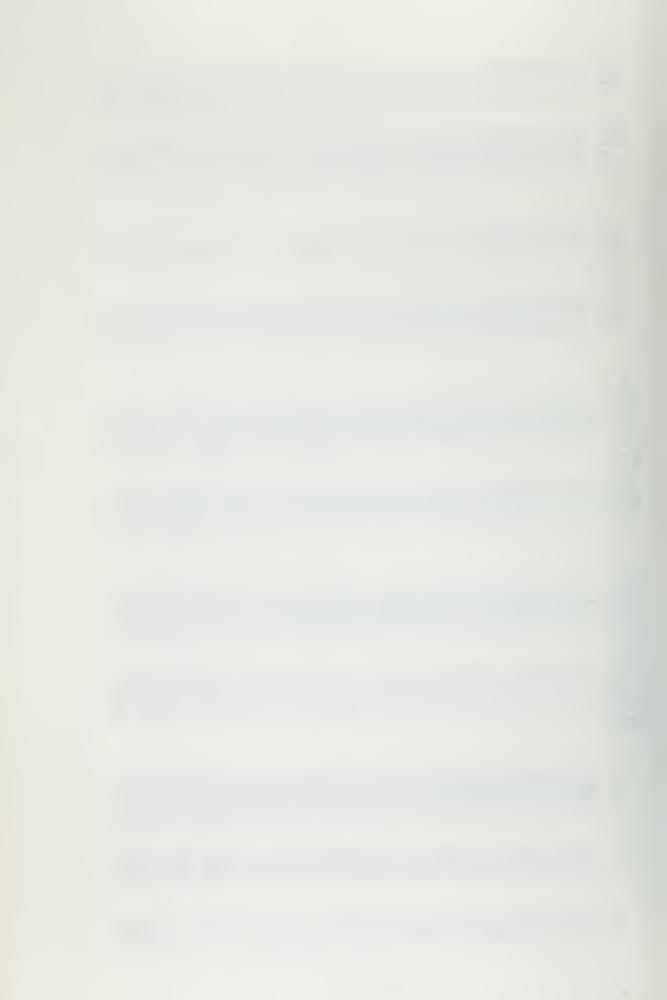
00	THE STATE OF THE S
DS SL	Р
00	
= Zn	В В В В В В В В В В В В В В В В В В В
≽# ≅S	OF 000000000000000000000000000000000000
	王 王
000000000000000000000000000000000000000	ADD ALCON A LONG
DM1 DM2 VE =	B 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
9 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	10-11-1 CS TABLES D1STES 667287-1 109119-1 1091182-1 109119-1 109011-
FOR IDNO DKG1 = DKG2 = IREF = DMAX =	BAAVA A STANDARD A STA
COEFFICIENTS = 2.0639992 = 0.0	AL AL MANAGE AND A WAS A
11 11	- wwww444nnnnaaaawwww444nnnnaa 4 oonnoonnoonnoonnoonnoonnoo 0 0000000000
WEAPON CFORMI CFORMI ITYPE IBOTH	©



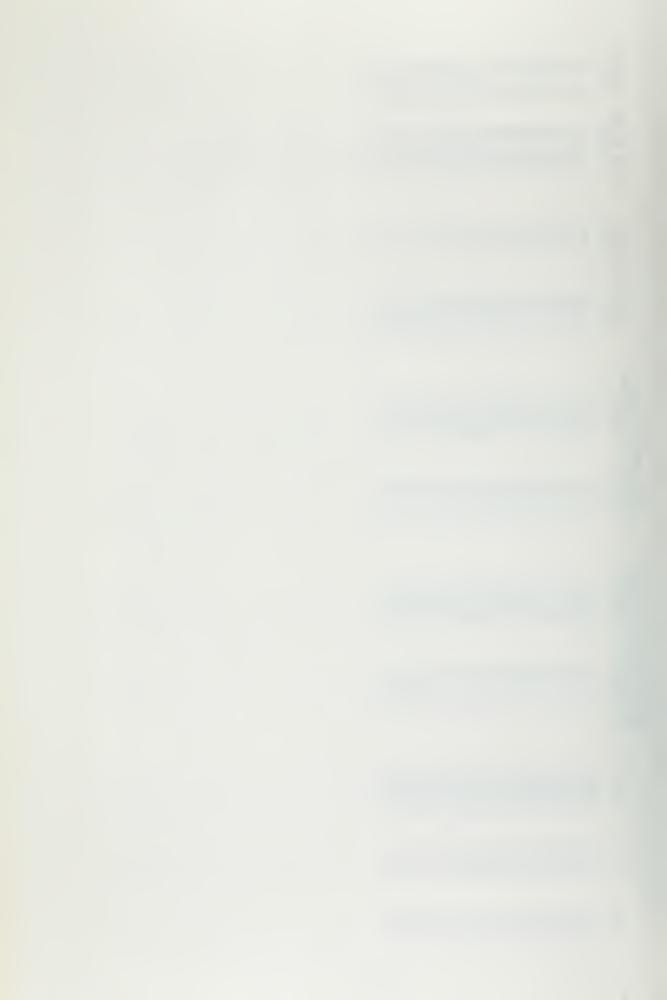




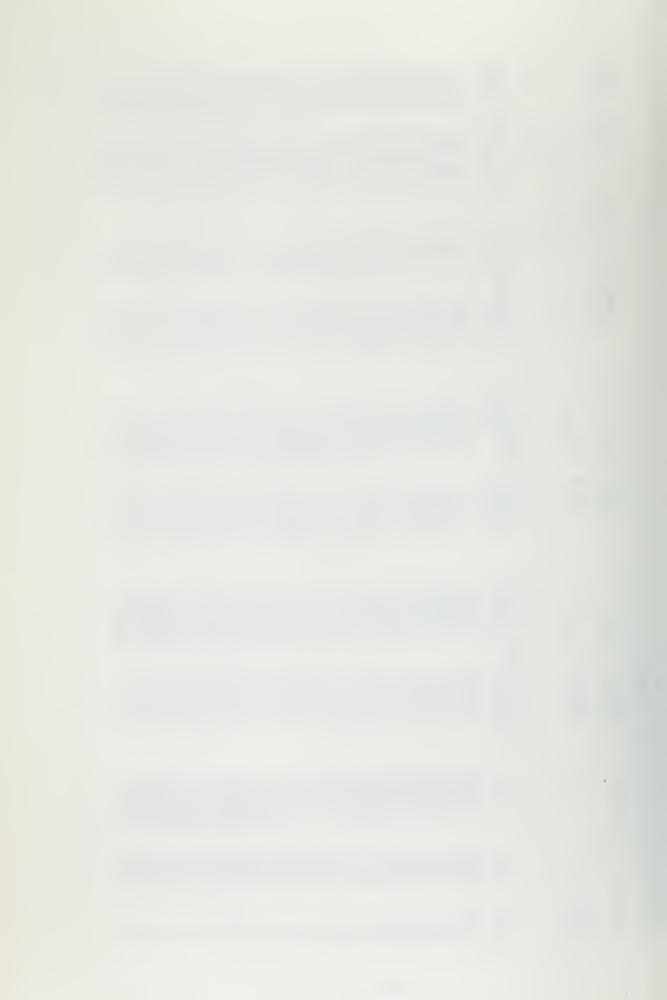




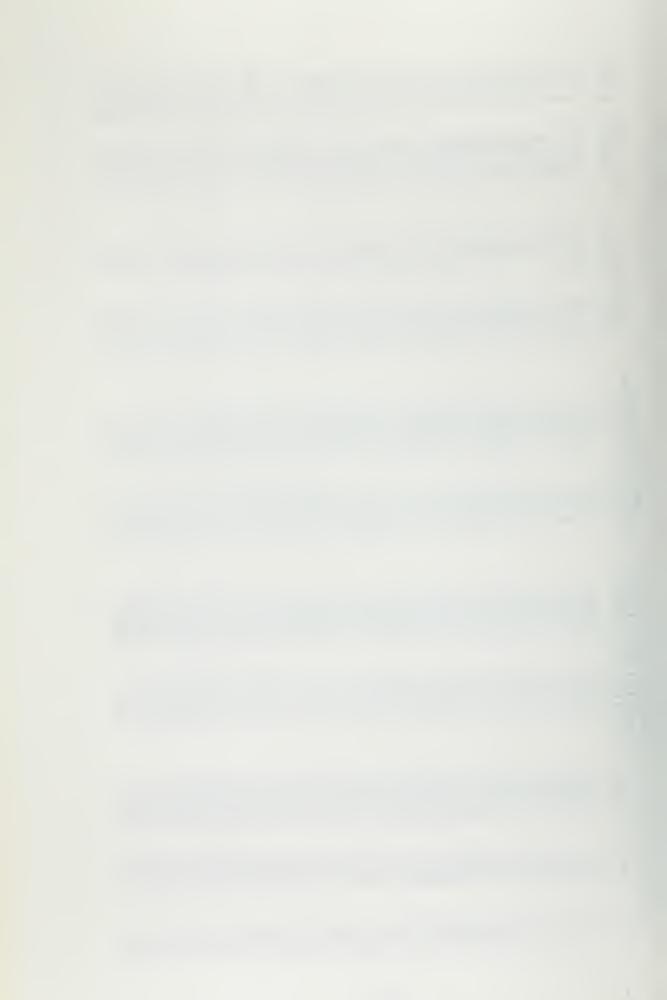
PER CENT ERROR TIME DIST	10000000000000000000000000000000000000
RENCES	0.000000000000000000000000000000000000
DIFFEF TIME	141717171111111111111111111111111111111
ODIFIED ALGORITHM DIST	1152 1152 1152 1153 1153 1153 1153 1153
BOEING /	2 1 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
STABLES DIST	110672. 1180672. 190677. 1912. 578073. 641173. 67173.
NAVAIR 01 BALLISTIC TIME	1 1 1 1 1 2 2 1 2 1 2 2 2 2 2 2 2 2 2 2
ALT	6 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
TAS	00000000000000000000000000000000000000
DEG	11111111111111111111111111111111111111

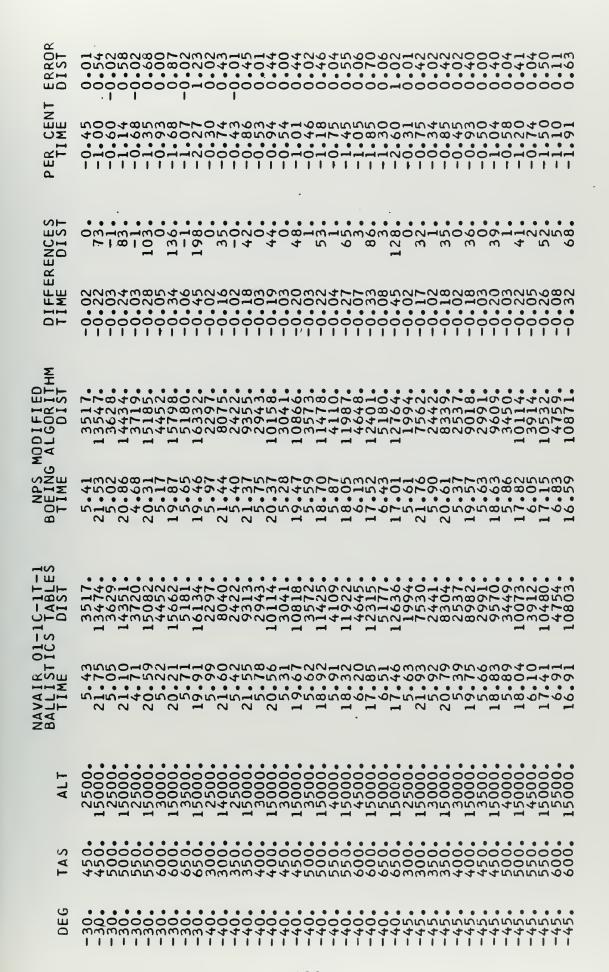


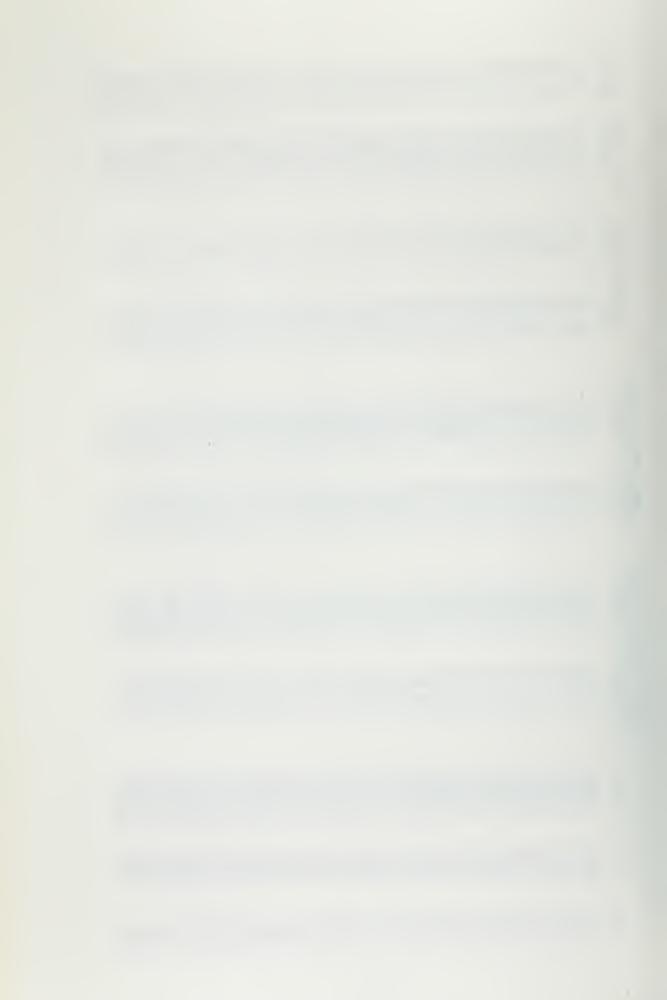
00		ERROR DIST	10101010000000111111101000000000000000
0 • 0 SL =		PER CENT TIME	
- ZNUZ = NNUZ		FF ER ENCES ME DIST	002 005 005 005 005 005 005 005 005 005
7L		HM MH	
00 0	= 3.00 MODIFIED	ALGORIT DIST	2 2 2 2 1 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2
MA 1	NPS	BOEING	1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
10	,00 1C-1T	CS TABLE DIST	2 2 2 2 1 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2
FOR IDNO DKG1 DKG2	DMAX DMAX AVAIR	BALLISTI TIME	2 3 3 8 8 8 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
EFFICIENTS 1.4931993 0.0		ALT	
PON CO		TAS	wwww4444wwww000ww00ww00ww00
ш шш і	1 BO	DEG	000000000000000000000000000000000000000



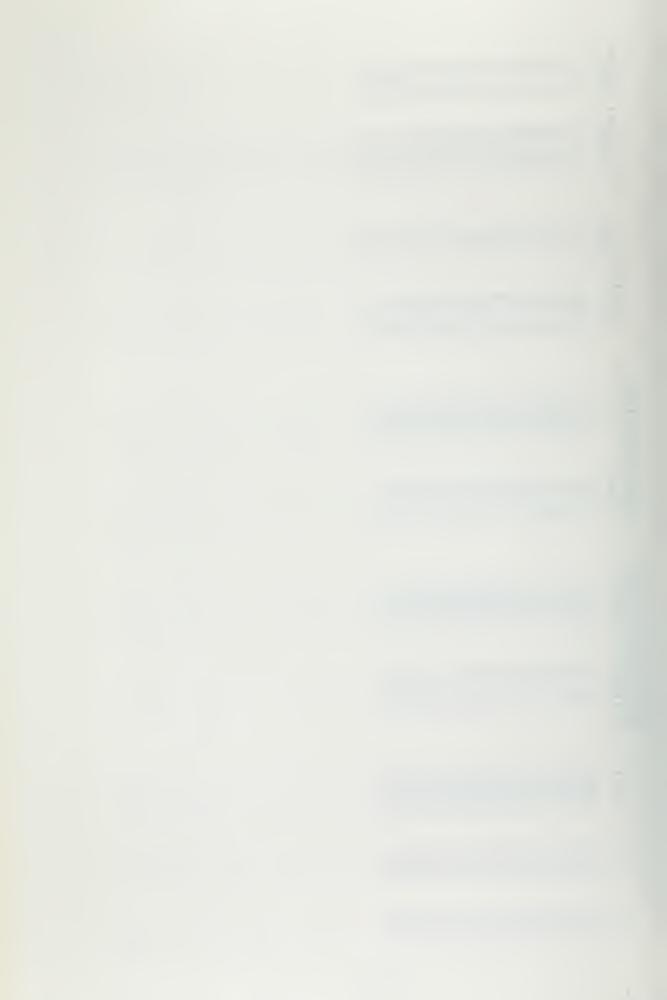
T ERROR DIST	00000000000000000000000000000000000000	
PER CEN		
RENCES DIST	6 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2	
DIFFE	00000000000000000000000000000000000000	
MODIFIED ALGORITHM DIST	2 2 1 1 1 2 2 2 1 1 1 1 2 2 2 2 2 2 2 2	
BOEING TIME	8 H I I I 2 2 2 2 I I I 2 2 2 2 I I I 2 2 2 2 I I I 2 2 2 2 I I I 2 2 2 2 2 I I I 2 2 2 2 2 I I I 2 2 2 2 2 I I I 2 2 2 2 2 I I I 2 2 2 2 2 2 I I I 2 2 2 2 2 2 I I I 2 2 2 2 2 2 I I I 2 2 2 2 2 2 I I I 2 2 2 2 2 2 2 I I I 2	
1-1C-1T-1 CS TABLES DIST	2 2 1 1 1 1 3 2 2 2 2 3 3 3 3 3 3 3 3 3	
NAVAIR O		
ALT		
TAS		
DEG	00000000000000000000000000000000000000	



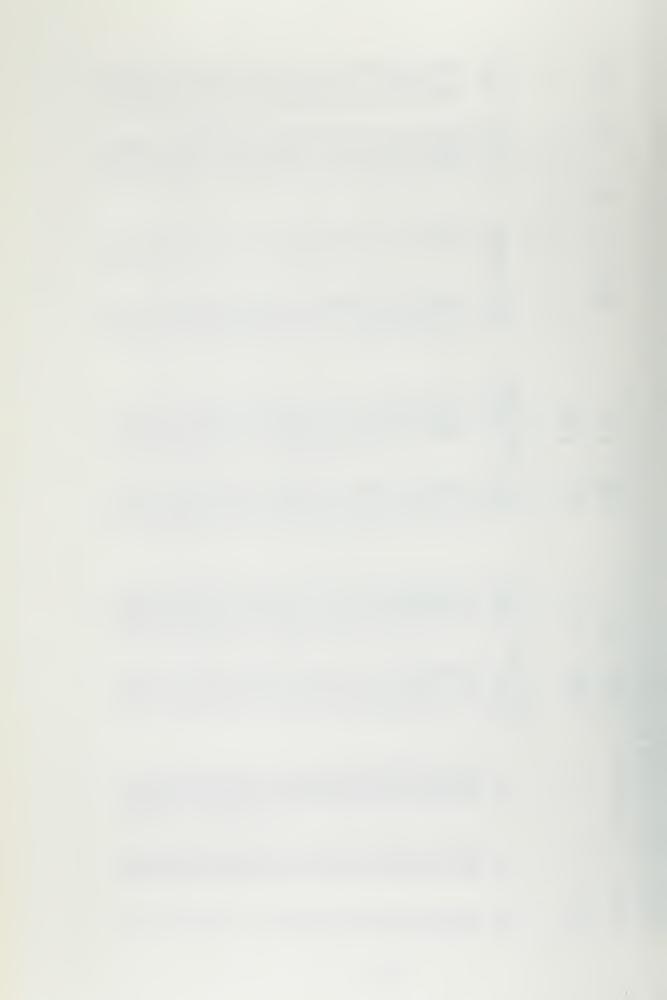




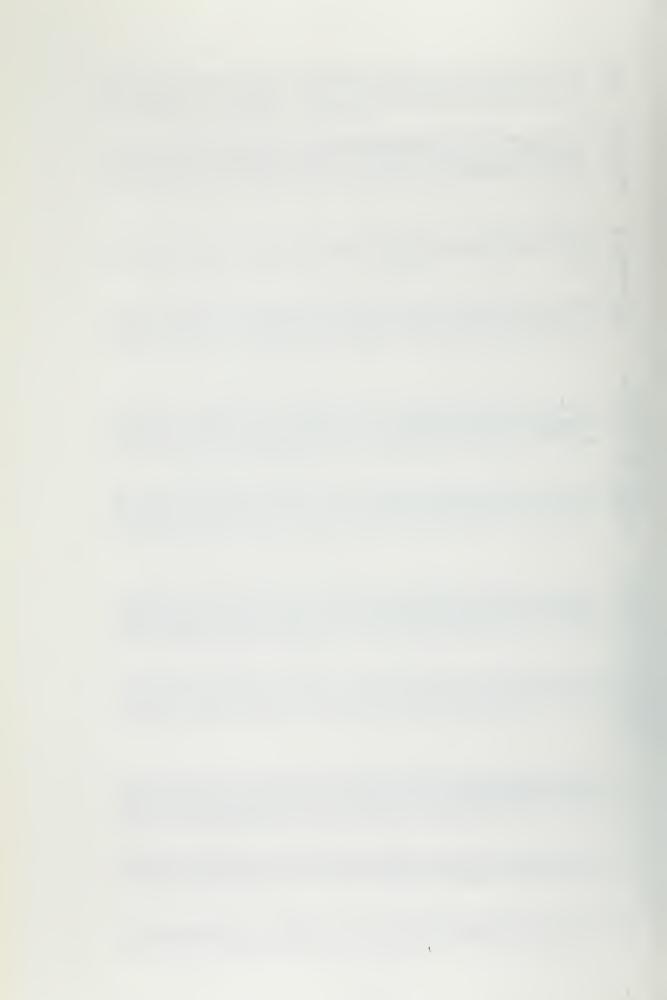
T ERROR DIST	00000000000000000000000000000000000000
PER CENT TIME	
RENCES DIST	10 100 1100 200 200 200 200 200 200 200
DIFFE	00000000000000000000000000000000000000
MODIFIED ALGORITHM DIST	1152 1172 1172 1172 1408 1409 1400 1400 1400 1400 1400 1400 1400
NPS BOEING TIME	16.05 16.05 17.05 17.05 17.05 17.05 17.05 18.05
-1C-1T-1 S TABLES DIST	1152 1100 1100 120 120 120 120 120 120 120
NAVAIR 01 BALLISTIC TIME	10000000000000000000000000000000000000
ALT	1
TAS	00000000000000000000000000000000000000
DEG	11111111111111111111111111111111111111



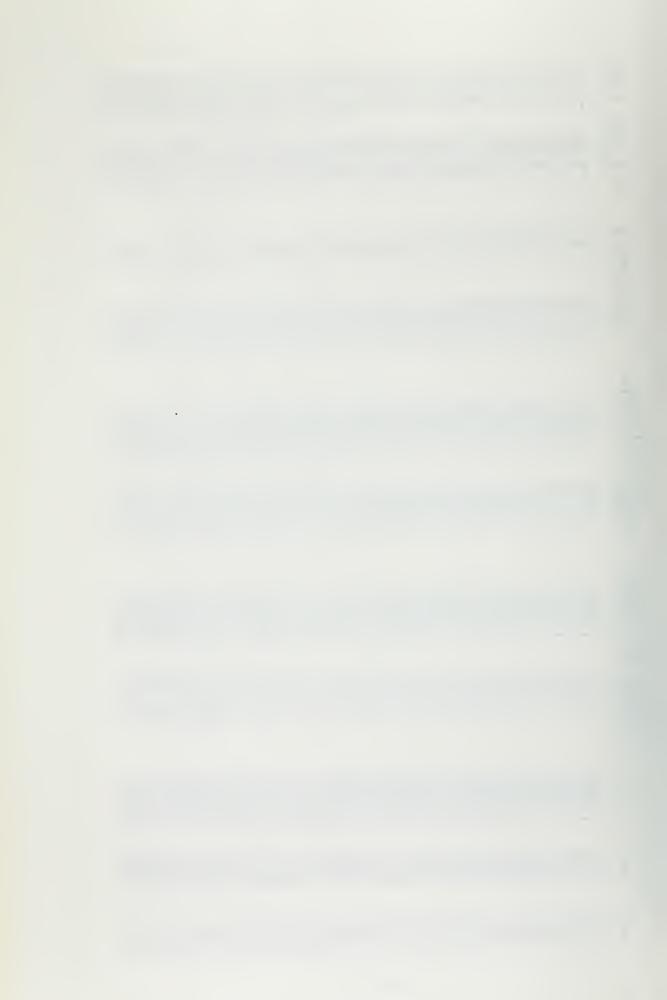
00		ROR ST	070101000001m170101010101010101010101010
00		ER DI	000000000000000000000000000000000000000
S_1		F.	
ON		O I M	000101000000010100000000000000000000000
		ER 1	000000000000000000000000000000000000000
••		٥	
00	•		
		SHS	14441101111100401111100041010
H		ENC	
701		E S	
>r XX		I A H	000000000000000000000000000000000000000
		Q -	000000000000000000000000000000000000000
		N.	
		ED IT ST	
00	00	IFI GOR DI	4800018000170040404000004047770 000000000000000
00	0	0 D A L	2 2 2 2 1 1 2000000
	" "	SZ m S	4m0000404NWN4L4NW04010L1LNWN9W4
WW.	N T T	I M	のころにはあるようは、これのこのでのできることのできる。
		80 T	3 3 3 3 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5
		-S	
		11 - BLE ST	
00	00	CTA	41000000000000000000000000000000000000
111	La	1-1 cs	
11 11	11 11	1 1	ら4190715947500000000000000000000000000000000000
IDNO KG1 KG2	A A A X	A I I S	00000000000000000000000000000000000000
	HO	> J F	
FOR		Zα	
15			• • • • • • • • • • • • • • • • • • • •
IEN 099		ALT	000000000000000000000000000000000000000
1C1		4	
COEFFICIENT = 1.3430996 = 0.0			
CO = 1	ł	AS	
N HO	Ш <u>Т</u>	 	るるとは、これでは、これでは、これでは、これでは、これでは、これでは、これでは、これで
A DO RR	7 O T	9	••••••••
3 00	11 18	DE	000000000000000000000000000000000000000



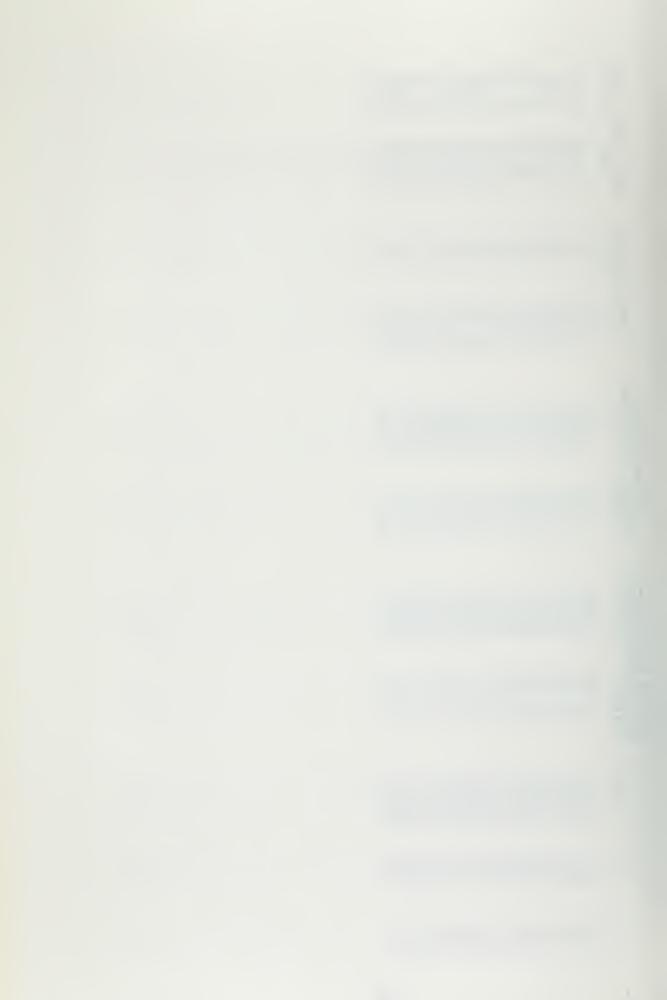
ERROR	00000		000000		000000000000000000000000000000000000000	000000
PER CENT TIME	00000 00000 00000 00000	000000 000000 01010 0142 0145	9mmmNm0	20000000000000000000000000000000000000	11111111111111111111111111111111111111	000000
RENCES DIST						
DIFFE	00000					000000
MODIFIED ALGORITHM DIST	るというできることのこの	44044 600400 7100900	とりとりていると	44447 104447 104447	14219 14219 1843647 1948828 19780 19780 19780	2390450 23906450 539081120
NPS BOEING TIME	でしてまり	4000001 8704810	らまるで り 4 る で の の の の の の の の の の の の の	0457-005- 0801-07-	2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2	できるととは
STABLES DIST	45-950	440 440 440 400 400 400 400 400 400	くしょうりょうしょくしょうしょうしょうしょうしょうしょうしょうしょうしょうしょうしょうしょうしょうし	107870 107470	1442223 1448669 1449669 144900 144900	2000000
NAVAIR 01 BALLISTIC TIME	01mm0	4979891 1686078 2184078	できららららっ り 4 ~ ~ で の i	7306736- 7301108-	7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7	で こ の に の に に の に に の に に に に に に に に に に に に に
ALT	000000	0000000 000000000000000000000000000000	0000000	00000000000000000000000000000000000000	77777777777777777777777777777777777777	000000000000000000000000000000000000000
TAS	00000	000000	0000000			000000
DEG	-4			SUNDINGE	00000000000000000000000000000000000000	Immmmmm



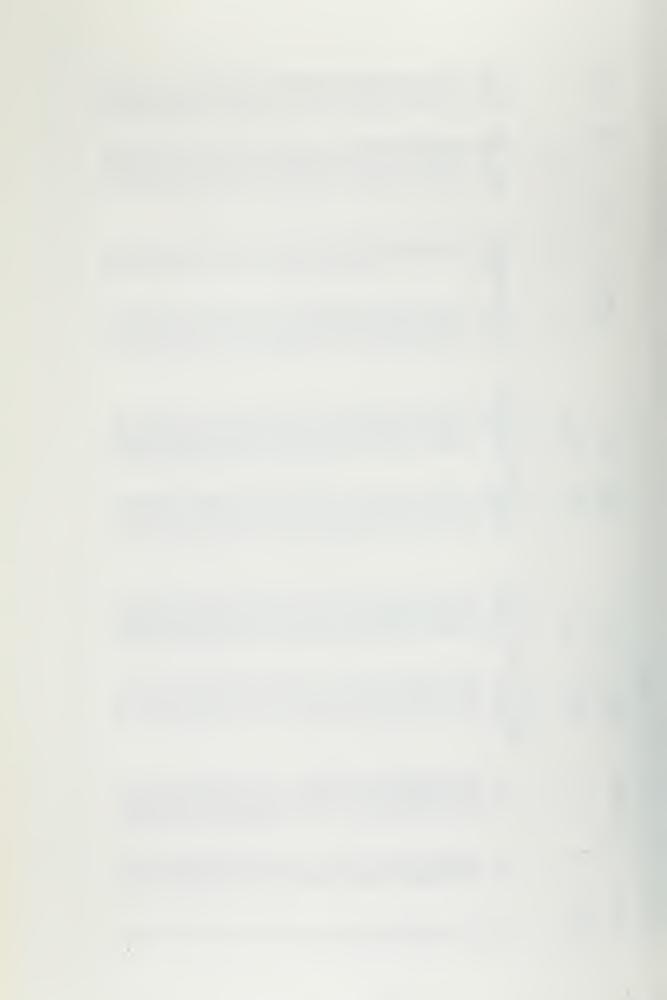
PER CENT ERROR TIME DIST	00000000000000000000000000000000000000	0.30 -0.0
RENCES	1 1 1 1 1 1 1 1 1 1 1 1 1 1	
DIFFE		00
Σ		
MODIFIED ALGORIT DIST	1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	476 089
NPS P BOEING TIME	00100000000000000000000000000000000000	က္မယ
HS		
-1C-1T- STABLE DIST	1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	906
NAVAIR OI BALLISTIC TIME	0.000000000000000000000000000000000000	6.5
	•	
ALT	22 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2	5500
TAS		00
DEG		44



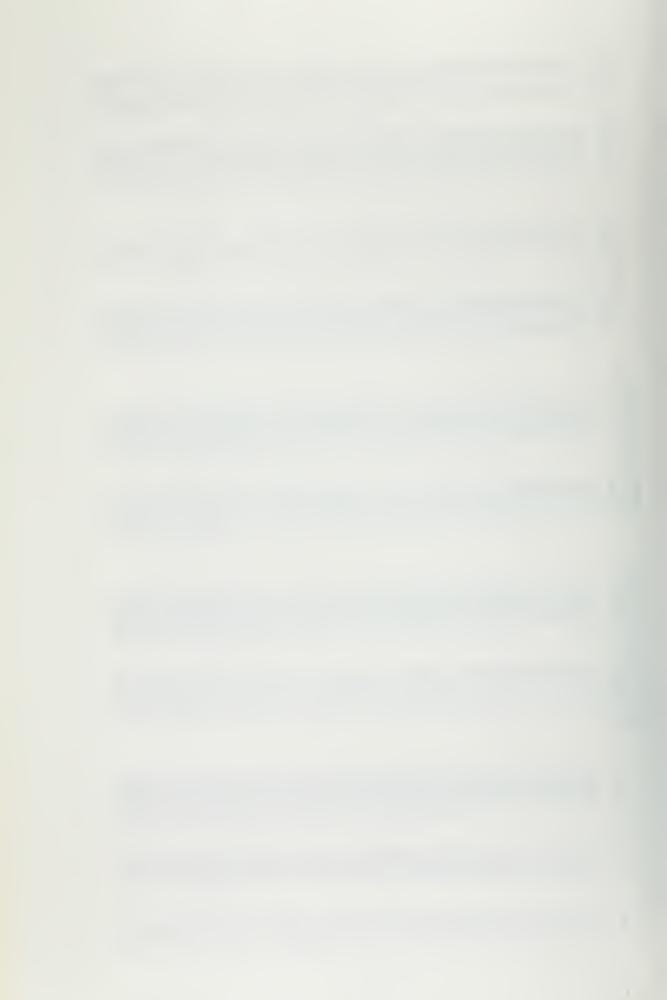
IT ERROR DIST	-00-100-100-100-100-100-100-100-100-100					000
PER CEN TIME	0000	-0-	000		000	00.0
RENCES	102.	10-1-	40			
DIFFE	000000000000000000000000000000000000000	000	000			000
_						
ODIFIED ALGORITHM DIST	5223. 11203. 1822.	400 400 400 400 400 400 400 400 400 400	81	4105 413	264	933
NPS M BOEING TIME	15.93 15.93 7.26	100c	7-1-	-F8,	85. 8.10	€
1-1C-1T-1 CS TABLES DIST	5228. 11205. 1822.	914 426 426	691	140 206 414	282 283 293	821 738 982
NAVAIR 01 BALLISTIC TIME	15.95	١٣٥٥	7.7.	5.00 15.00	5.1	2010
ALT	15000.	14mm	0000	1000 1000 1000	500 500 850	000
TAS	0000 0000	0000	000	000	2000	SOWIN
DEG	449 2000		909	000	999	999



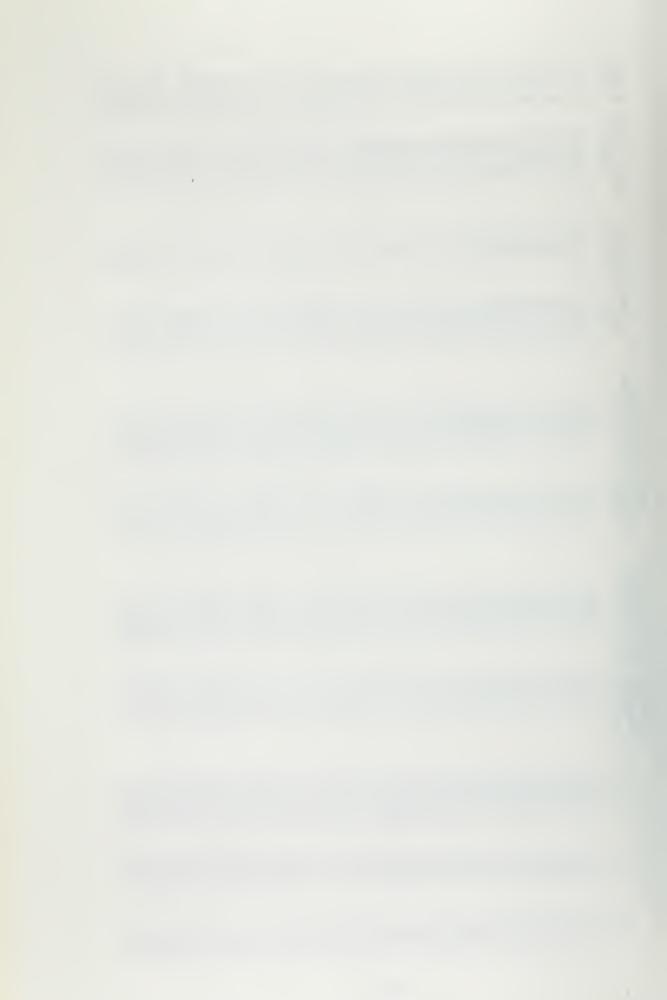
000		T ERROR DIST	00000000000000000000000000000000000000
DS SL		PER CEN	00000000000000000000000000000000000000
00			
= 70		RENCES	
≥n £N		DIFFE TIME	00000000000000000000000000000000000000
		Ĭ	
• •	0 0 0 0 8	MODIFIED ALGORITH DIST	2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
M2 M2	NO TE	NPS BOEING	1 3 3 3 2 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1
		ES.	••••••
	1 5.00	S TABL DIST	2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
IDNO 1 KG1 = KG2 =	" "	R 01 STIC	######################################
FOR	IRE OMA	NAVAI BALLI TIM	
COEFFICIENTS = 1.2099991 = 0.0		AL T	11 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
		TAS	www.4444\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\
EAP FOR FOR	I TYPE I BOTH	DEG	000000000000000000000000000000000000000



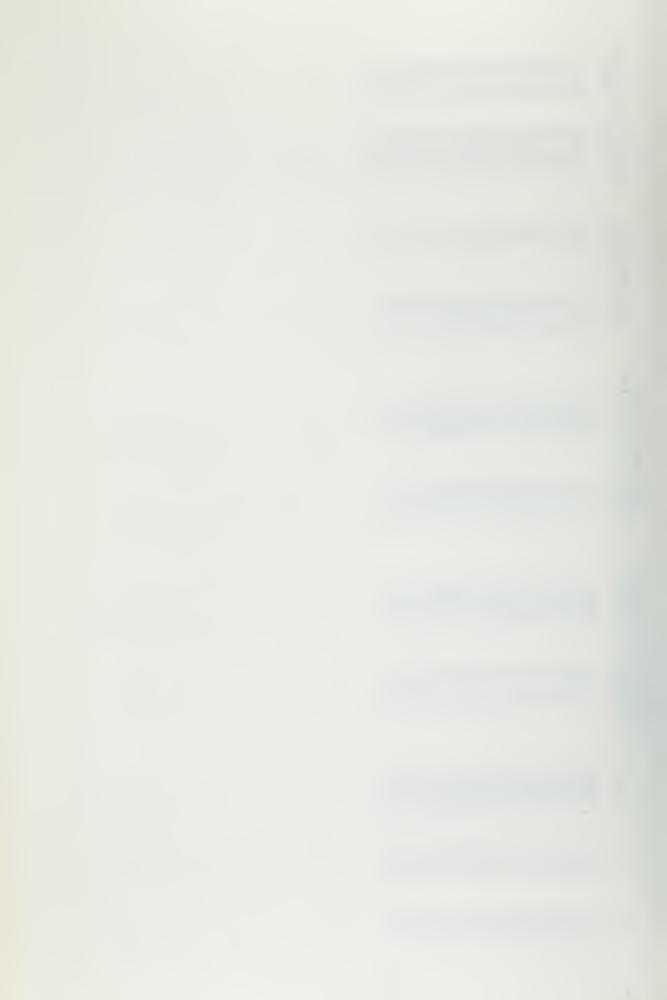




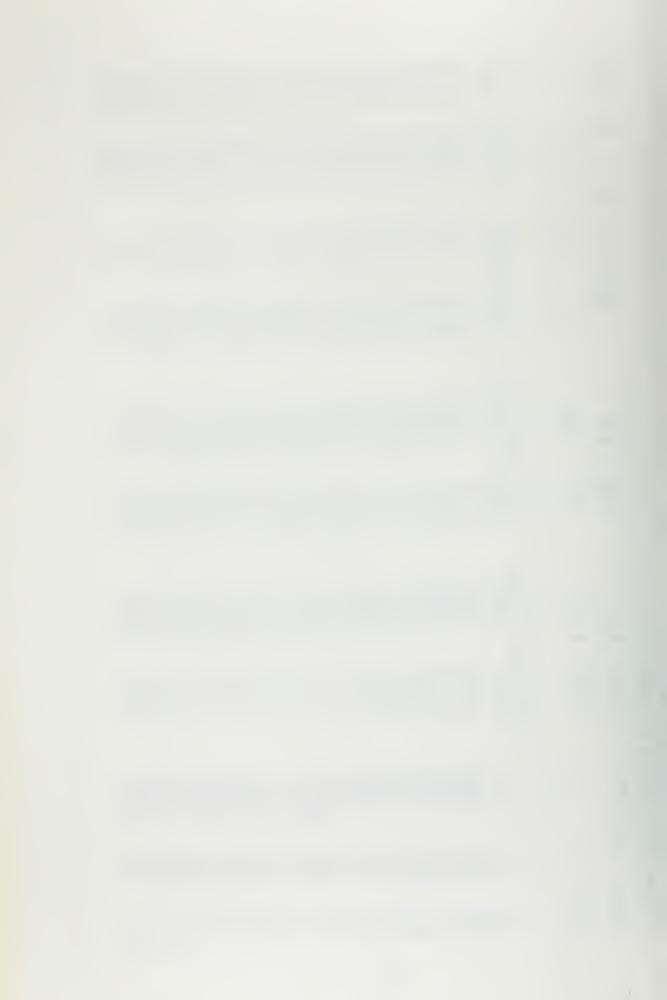
PER CENT ERROR TIME DIST	00000000000000000000000000000000000000
RENCES	
DIFFE	10000000000000000000000000000000000000
MODIFIED ALGORITHM DIST	1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
NPS BOEING TIME	0
1-1C-1T-1 CS TABLES DIST	1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
NAVAIR O BALLISTIO TIME	
ALT	\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
TAS	
DEG	

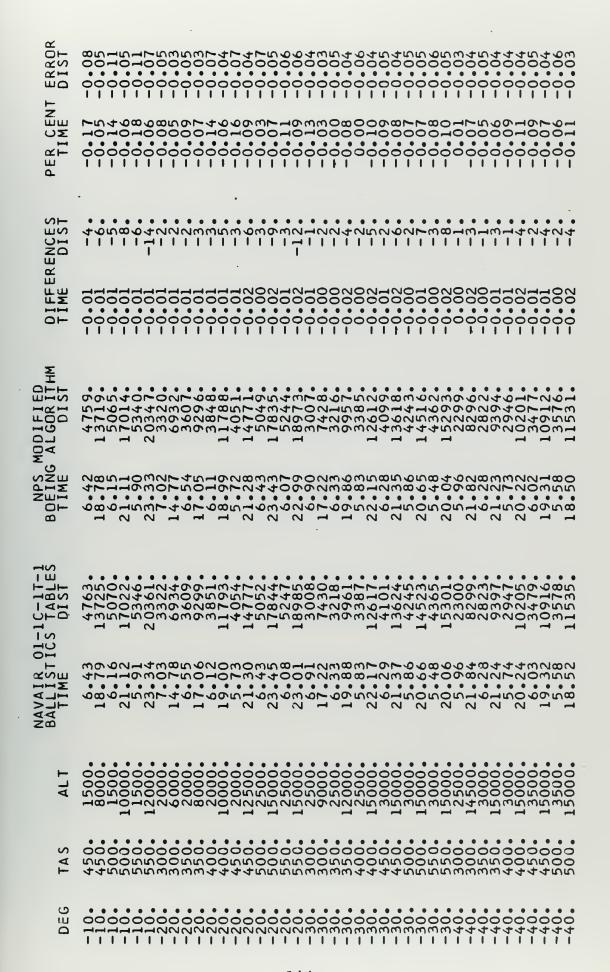


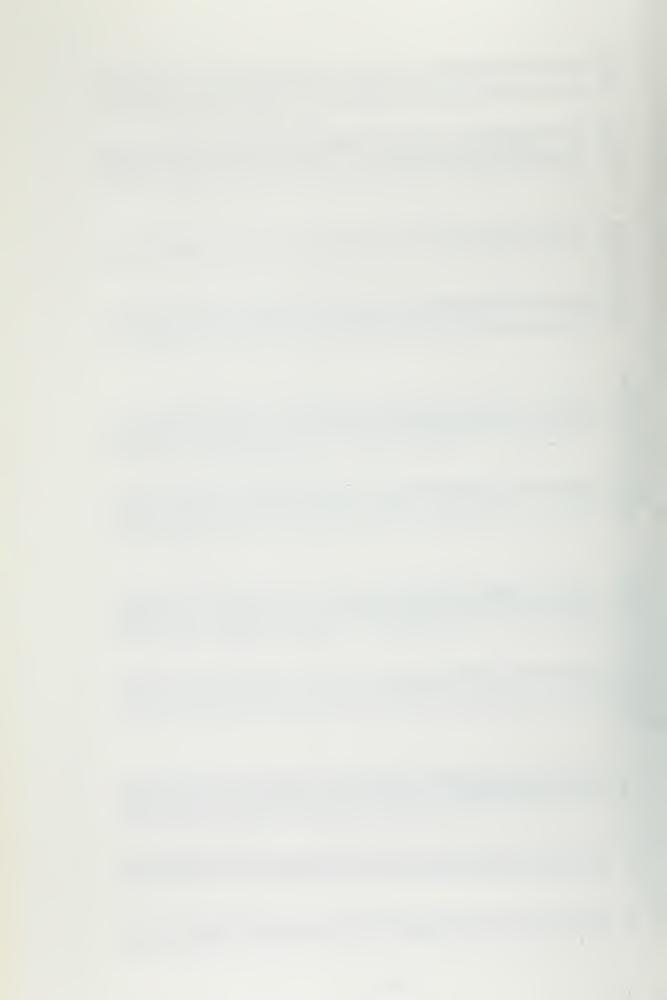
T ERROR DIST	000000000000000000000000000000000000000	
PER CEN TIME		2000000 4wnnv-
RENCES DIST	1 in 0 1 in 0 in 0 in 0 in 0 in 0 in 0 i	1 400-1 800-
DIFFE	441000000000000000000000000000000000000	000000
MODIFIED ALGORITHM DIST	115222 1408224 14092226 1409144 16093 1609	978765t 978760t 947840t
NPS P BOEING TIME	15.00 20.26 20.26 18.53 17.21 17.21 17.21 16.69	00000000000000000000000000000000000000
-1C-1T-1 S TABLES DIST	11222 11222 1222 1222 1222 1222 1222 1	478780P 6787780P
NAVAIR 01 BALLISTIC TIME	15.00 10.00	27-48-46-48-48-48-48-48-48-48-48-48-48-48-48-48-
ALT	00000000000000000000000000000000000000	$v_{\mathbf{L}}$ $v_{\mathbf{R}}$ $v_{\mathbf{L}}$ $v_{\mathbf{L}}$
TAS	00000000000000000000000000000000000000	0000000
DEG	111111111 444040404040 0000000000	000000



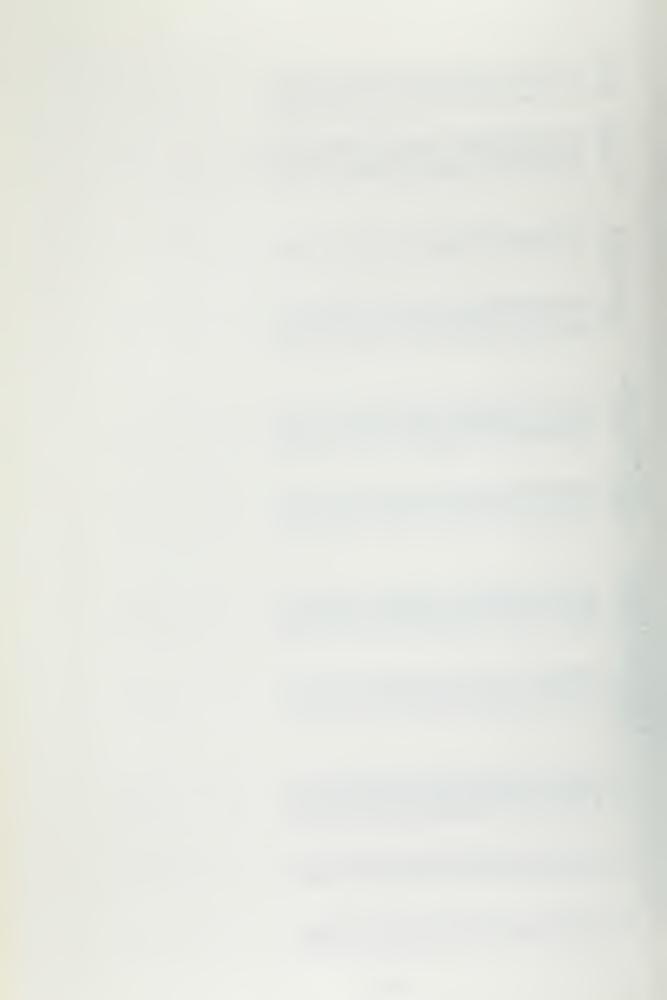
	000		T ERROR DIST	00000000000000000000000000000000000000
	DS SL		PER CENT	00000000000000000000000000000000000000
	00			·
	= 71		RENCES	
	D N N		DIFFE	
			¥	
	0.0	3.00	ODIFIED ALGORIT DIST	2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
	DM1 =	VE = DTI =	NPS M BOEING TIME	1 3 3 3 3 3 1 1 1 1 1 1 1 1 1 1 1 1 1 1
8	00	•	S TABLE DIST	2 2 2 2 2 3 2 3 3 3 3 3 3 3 3 3 3 3 3 3
10 1	11 11	11 11	T IC	
FOR IDNO	DKG1 DKG2	IREF DMAX	NAVAIR BALLIS TIME	
NTS	. 00		Jan	000000000000000000000000000000000000000
IN COEFFICIENT	0000000		AL	
	$\begin{array}{ccc} 1 & = & 1 \\ 2 & = & 0 \end{array}$		TAS	
WEAPON	CFORM	I TYPE I BOTH	DEG	



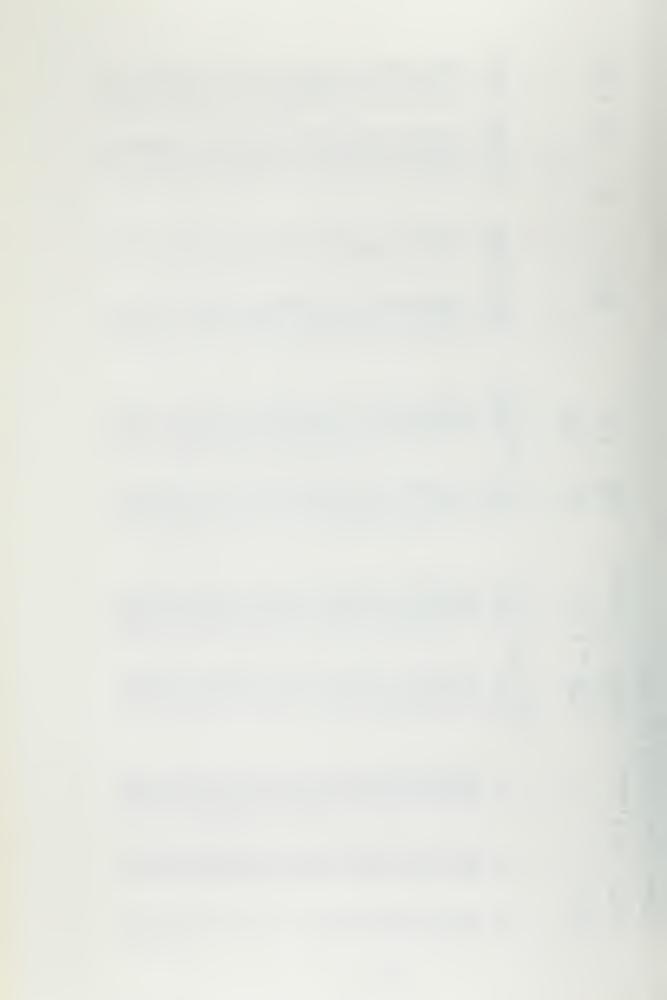




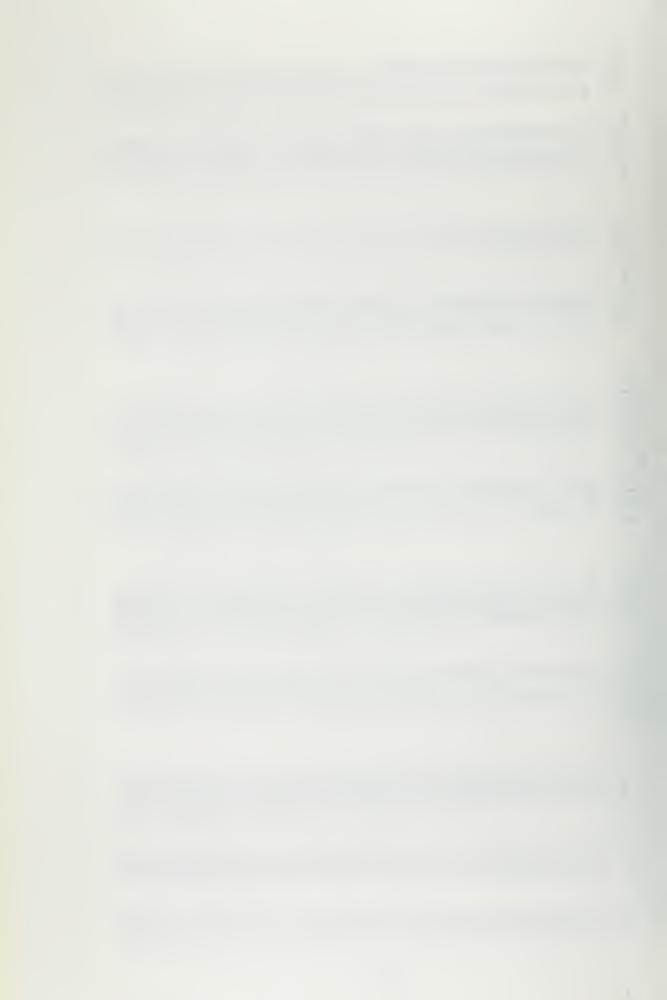
		-	
T ERROR DIST	00000000000000000000000000000000000000		00000000
PER CEN TIME	00000000000000000000000000000000000000		00000000
RENCES	111111111 00000444	111111	
DIFFE	000000000000000000000000000000000000000	0000000	00000000
MODIFIED ALGORITHM DIST	120114. 123119. 75919. 24441. 28871. 29052.	04100800 400181	44841454 600650000
NPS BOEING TIME	らてもようのものら	1000000	412161118 961661611
1-1C-1T-1 CS TABLES DIST	12062 12062 12319 12319 12594 12905	90800 9080 9080 9080 9080	448717459 WONOWONOR
NAVAIR O BALLISTI TIME	2210 200 200 200 200 200 200 200 200 200	50000000000000000000000000000000000000	90-100-00
ALT	4nwnwnwnwn 000000non 000000000	v4v4v4v4 0000000 00000000	000000000
TAS	nnwwww444	00000000	NOONNOONN
0EG	44444444 00000000000000	44444000 000000000	000000000



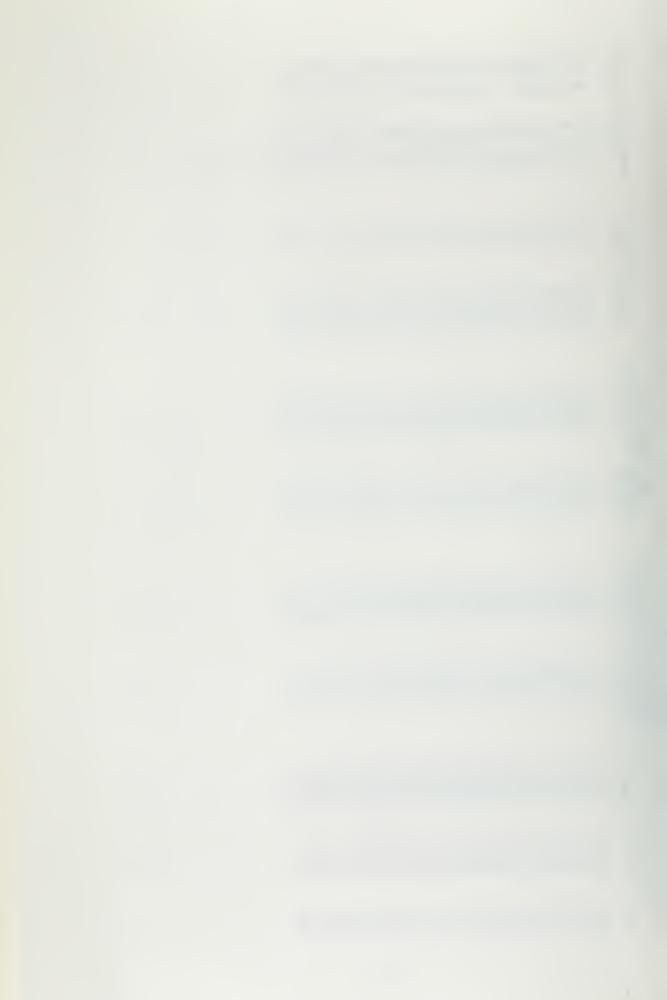
0.0 DS = 0.0 0.0 SL = 0.0	PER CENT ERROR IN MEDICAL POLICY PER CENT PER CE
KWUZ = EN =	TIME ENCES TIME ENCES TIME ENCES TO 00000000000000000000000000000000000
DM1 = 0.0 DM2 = 0.0 VE = 0.0 DTI = 3.00	BDEING ALGORITHM TIME DIST 16.77 8082. 16.77 10.27 10.27 10.27 11.68 12.40 12.40 12.40 12.40 12.40 14.911. 19.68 14.911. 19.68 14.911. 19.68 17.26 17.31 17.34 17.
FOR IDNO 14 DKG1 =0012230 DKG2 = 0.0 IREF = 1 DMAX = 5.00	NAVAIR 01-1C-1T-1 8 95
3.1159999 0.0 0.1	ALT
WEAPON COE CFORM1 = 3. CFORM2 = 0. ITYPE = -1.	



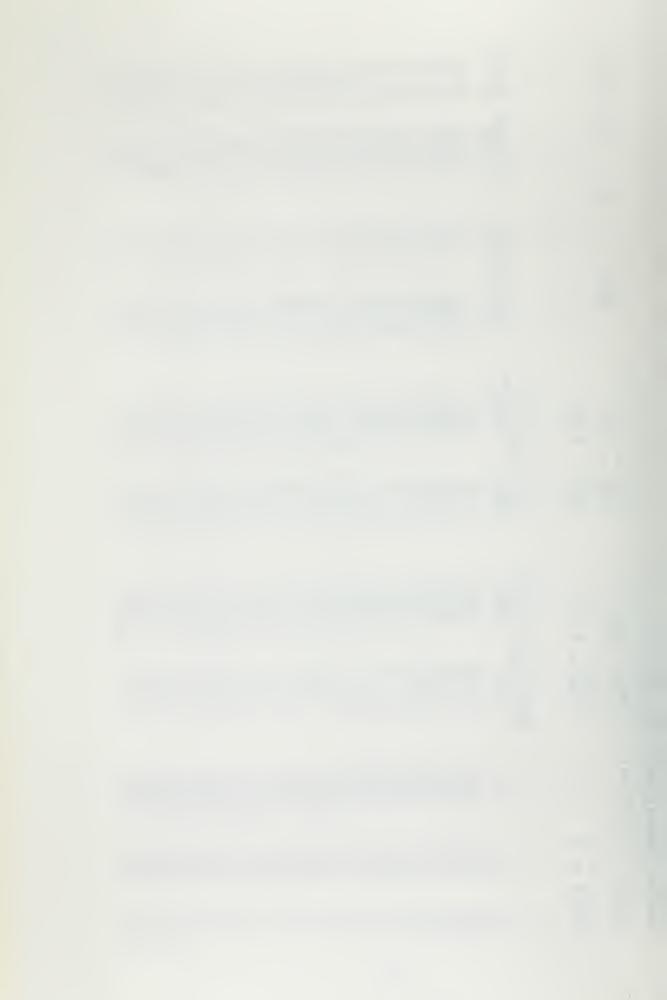
CENT ERROR	221212101201344492534445170000000000000000000000000000000000
PER	
ERENCES	
DIFF	
MODIFIED G ALGORITHM DIST	
NPS S BOEINC	1 2 2 1 1 1 2 2 2 2 1 2 2 2 2 1 1 2 2 2 2 2 1 1 2
Ol-1C-1T- ICS TABLE	1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
NAVAIR BALLIST TIME	1
ALT	
TAS	44NNNNWWWWWWA444NNNNWWWWWWA444NN NNOONNOONNOONNOONNOONNOONNOONNOO 00000000
DEG	

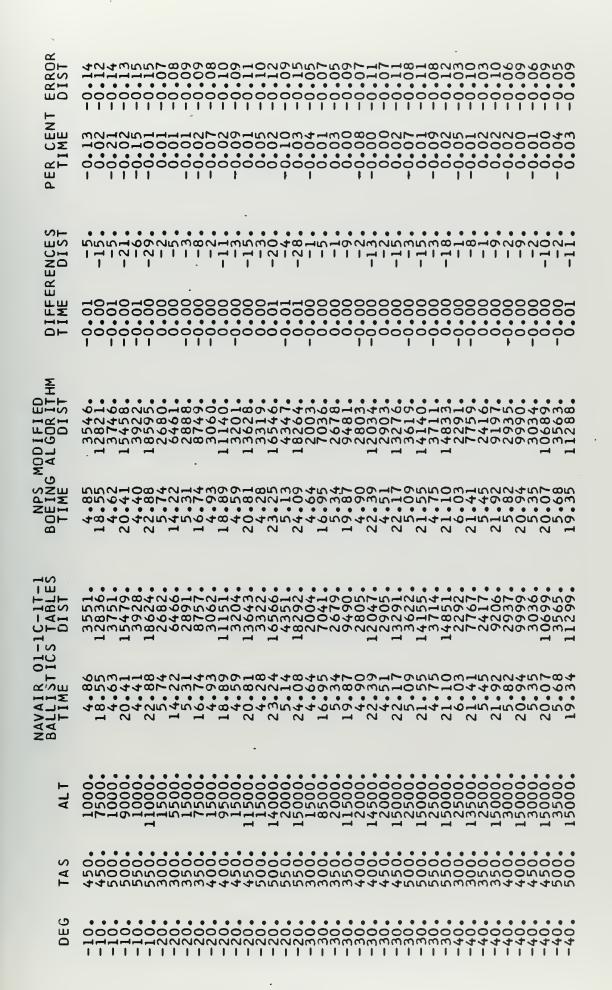


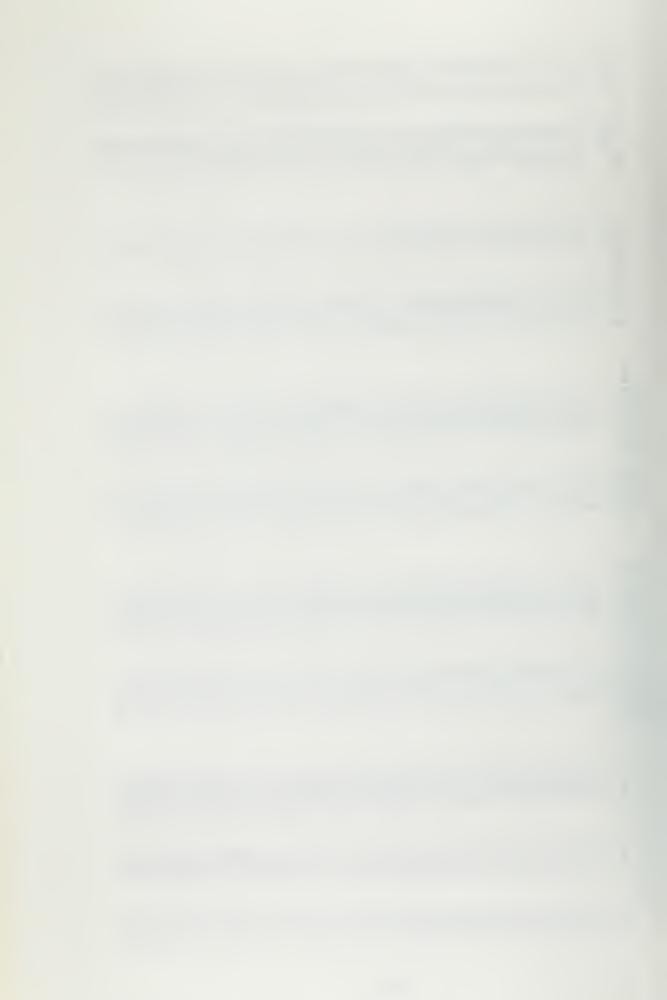
PER CENT ERROR TIME DIST	10000000000000000000000000000000000000	00000000000000000000000000000000000000	00.000000000000000000000000000000000000
FF ER ENCE S ME DIST	4019090909400		1111
10	0000000000	00000000	00000
MODIFIED ALGORITHM DIST	1118103 118003 129901 255888 25588 25588 25588 25588 25588 2558 2568 256	00480084 ~ 0	243100
NPS BOEING TIME		10000000000000000000000000000000000000	5-10-10
CS TABLES DIST	118511 118511 25536 82586 82586 82596 8447	1000mm00000000000000000000000000000000	10000r
NAVAIR O BALLISTI TIME	2 2 2 1 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2	78760700 7176451700	74677
ALT	11 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	v4w4w4www owooooo oooooooooooooooooooooo	20000000000000000000000000000000000000
TAS	00000000000000000000000000000000000000		00000 00000
DEG	00WWWWWWWWW	200000044	00000



DS = 0.0 SL = 0.0	R CENT ERROR TIME DIST	00000000000000000000000000000000000000
• •	P E	
= 7r	ER ENCES DIST	
ΣZ >L	DIFF	000000000000000000000000000000000000000
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	MODIFIED ALGORITHM DIST	2 2 2 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0
DM DM2 DTI	NPS BOEING TIME	1 1 3 8 8 8 8 1 1 1 1 1 8 8 8 8 8 1 1 1 1
15 : 0.0 : 0.0 : 3.00	1-1C-1T-1 CS TABLES DIST	2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
FOR IDNO DKG1 = DKG2 = IREF = DMAX =	NAVAIR O BALLISTI TIME	1
1C1ENTS 571991	ALT	\$\\ \text{\$\begin{array}{cccccccccccccccccccccccccccccccccccc
11 13	TAS	wwww444wwwwwwww444wwwwwwwww44 connoonwconnoonwconnoonwco
WEAPON CFORM1 CFORM2 ITYPE IBOTH	DEG	



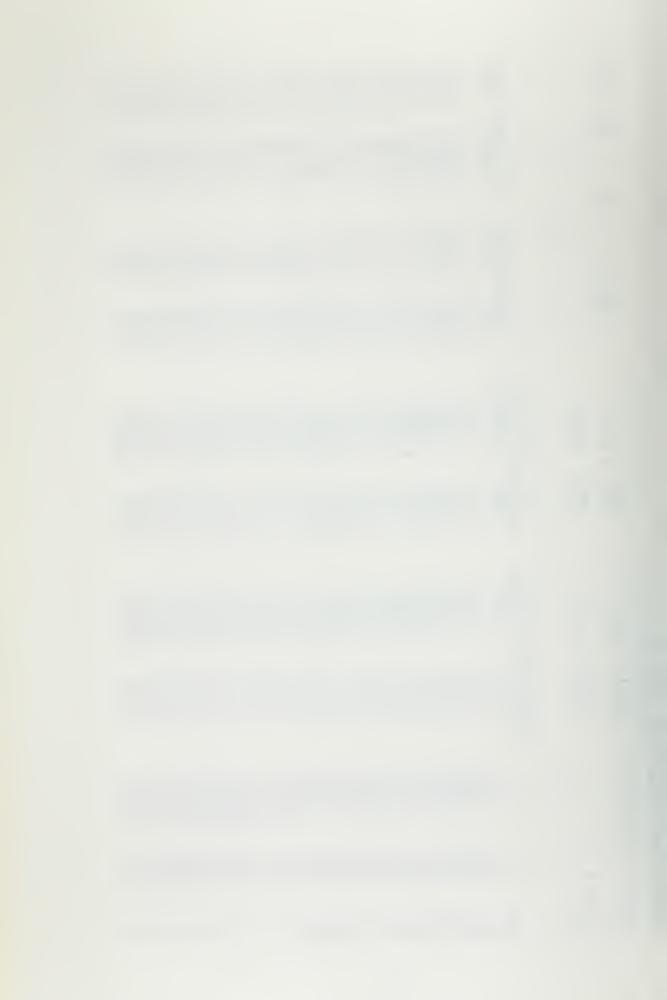


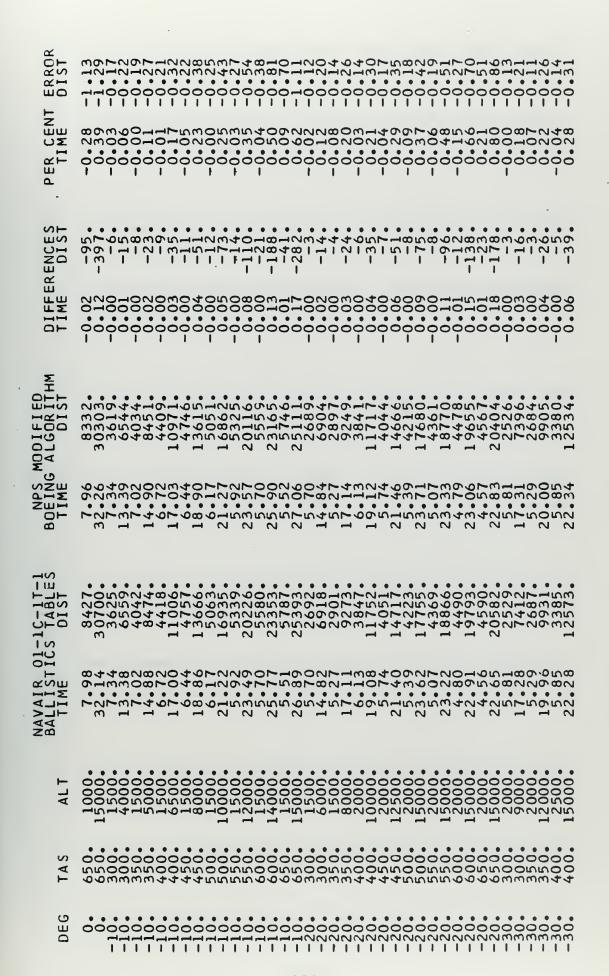


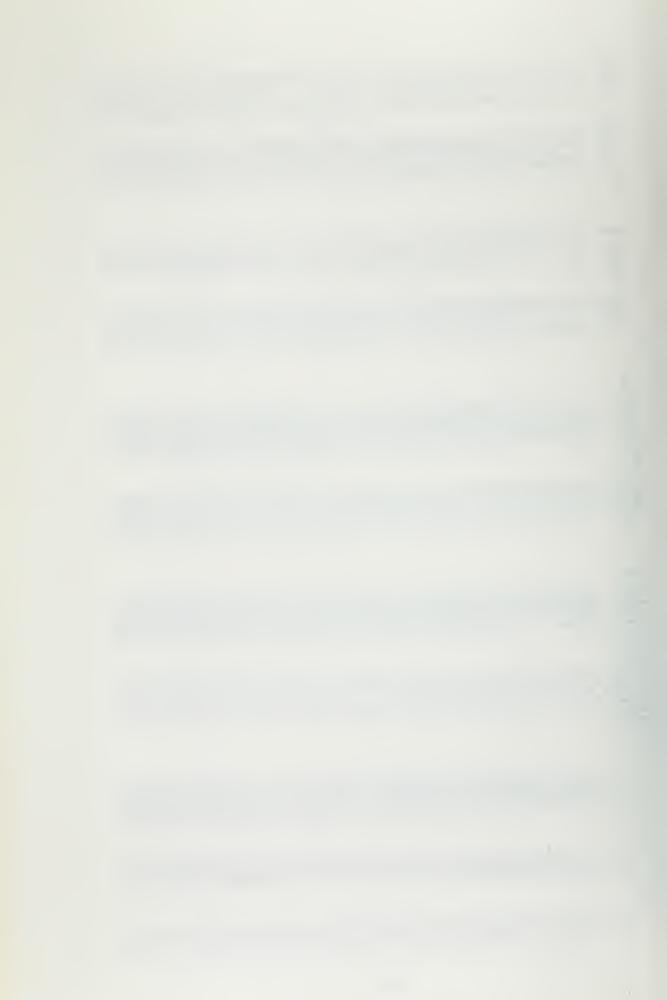
NT ERROR DIST	00000000000000000000000000000000000000	0
PER CE	00000000000000000000000000000000000000	
RENCES DIST		
DIFFE		
MODIFIED ALGORITHM DIST	1 100000000000000000000000000000000000	200
NPS BOEING TIME	2 2 2 1 1 2 2 2 2 2 2 1 1 2 2 2 2 2 2 2	ο ω
CS TABLES DIST	11100000000000000000000000000000000000	54
NAVAIR O BALLISTI TIME	2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2	5. 8.
ALT	4 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	000
TAS	wwwwwa444wwwwwwww4444www wwoowwoowwoowwoow oooooooo	200
DEG	1 1 1 1 1 1 1 1 1 1	09



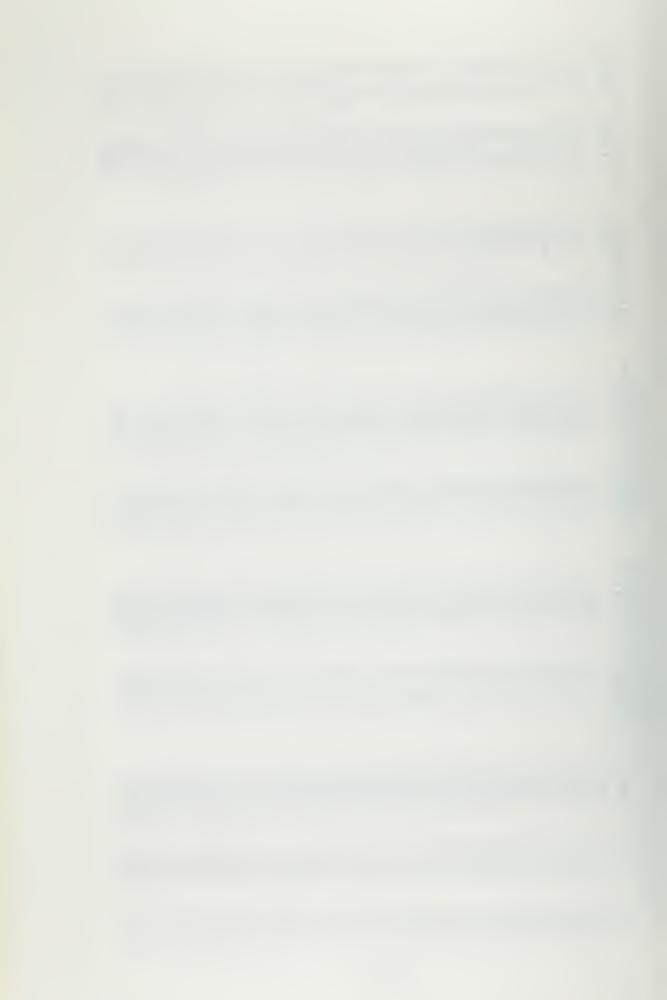
DS = 0.0 SL = 0.0	CENT ERROR IME DIST	### ### ### ### ### ### ### ### #### ####
••	PER	
" 7 "	RENCES DIST	21211111111111111111111111111111111111
>π ÆS	DIFFE	
	Σ	
000000000000000000000000000000000000000	1001FIED ALGORITHM DIST	2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
DM1 = DM2 = OTI = DTI =	NPS M BOEING TIME	3 3 3 3 51211111111111 1 8 8 8 8 8 8 8 8 8 8 8 8
16 : 0.0 : 0.0 : 1 : 5.00	CS TABLES DIST	2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
FOR IDNO DKG1 = DKG2 = IREF = DMAX =	NAVAIR BALLISTI TIME	1 3 3 3 3 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5
COEFFICIENTS = 1.6049995 · = 0.0	ALT	
11 11	TAS	wwww4444wwww0000ww00ww000ww00000000000
WEAPON CFORM1 CFORM2 ITYPE	DEG	000000000000000000000000000000000000000







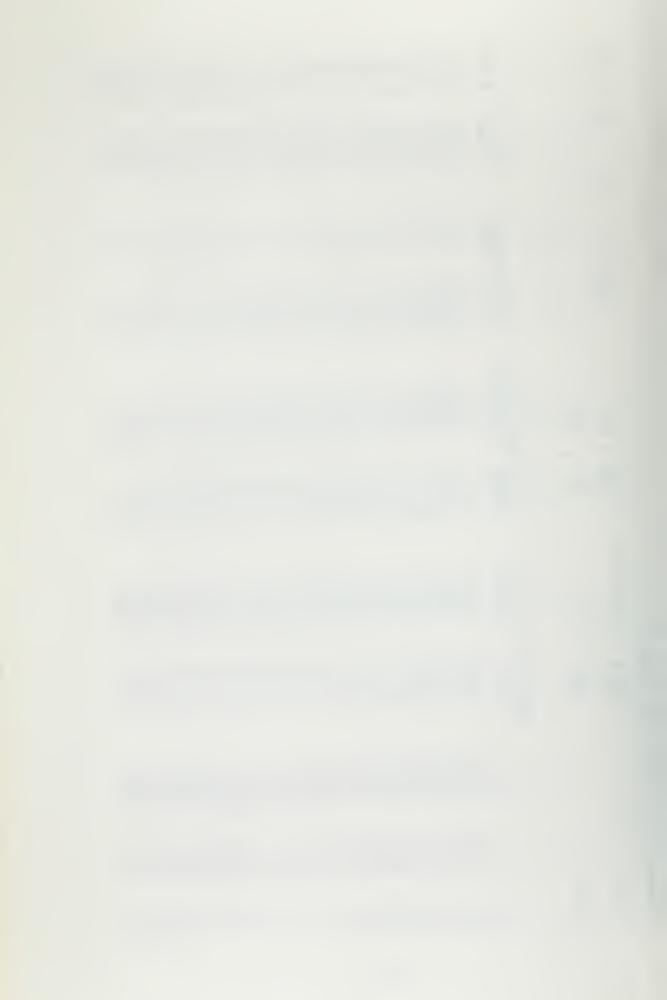
T ERROR DIST	00000000000000000000000000000000000000
PER CENTINE	00000000000000000000000000000000000000
RENCES	1414101810101010114141711181011181011414 4 14101010101010114141717111181010101010101010101010101010101
DIFFE	
돌	
MODIFIED ALGORITH DIST	1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
NPS P BOEING TIME	0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.
CS TABLES DIST	1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
NAVAIR OF BALLISTIC	0 0 0 0 1 1 2 2 2 2 1 1 1 2 2 2 2 1 1 1 2 2 2 2 1 1 1 2
ALT	NUNTAUMUMUMUMUMUMUMAAUAUMUMUMUMUMUMUMUMUMUM
TAS	44NNNN 9660 WWWWW 4444NNNN 6669 WWWWW 444NNNN 6660 WWOONNOONNOONNOONNOONNOONNOONNOONNOONNO
DEG	



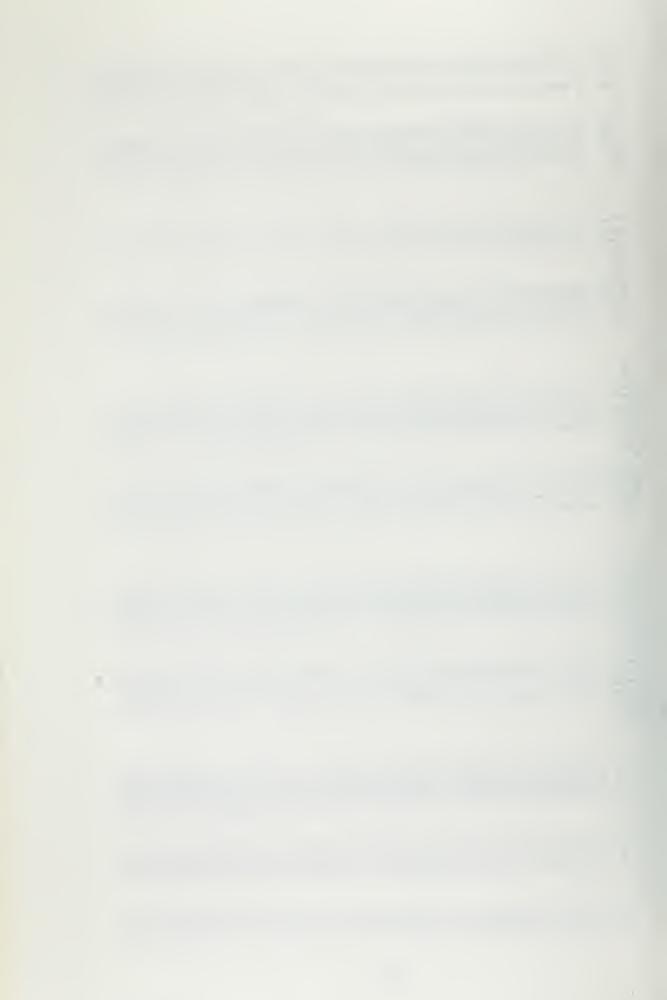
PER CENT ERROR TIME DIST	0.38 '-0.24 1.12 -0.51 0.02 -0.51	24 -0-1 06 -0-1	28 -0.2 12 -0.1	12 -0 0	57 -0.2	80 - 00	15 -0.4
RENCES DIST	11, 11, 11, 11, 11, 11, 11, 11, 11, 11,	-11- -2-	-1-	VMM		10-	-28.
DIFFE	00000				• •	• •	
MODIFIED ALGORITHM DIST	5215. 11147. 1821. 4961.	417	80	200 400 400	649	762	95
NPS BOEING TIME	7.07 16.13 7.27 20.20	780	7.8	97°6	5.7	-90	10
1-1C-1T-1 CS TABLES DIST	5228. 11205. 1822. 4972.	426 426 407	8169	140 206 414	630	187	96
NAVAIR 01 BALLISTIC TIME	15.95 20.27 20.15	780	7.7	5.70	-1	٥١٠٠	-6.
ALT	150000 150000 150000	500 500 500	200	500 500 500	000	500 500 500 500 500	500
TAS	00000000000000000000000000000000000000	000 000	500	200	200	000	200
DEG	1111 4400 0000	999	99	999	999	00	9



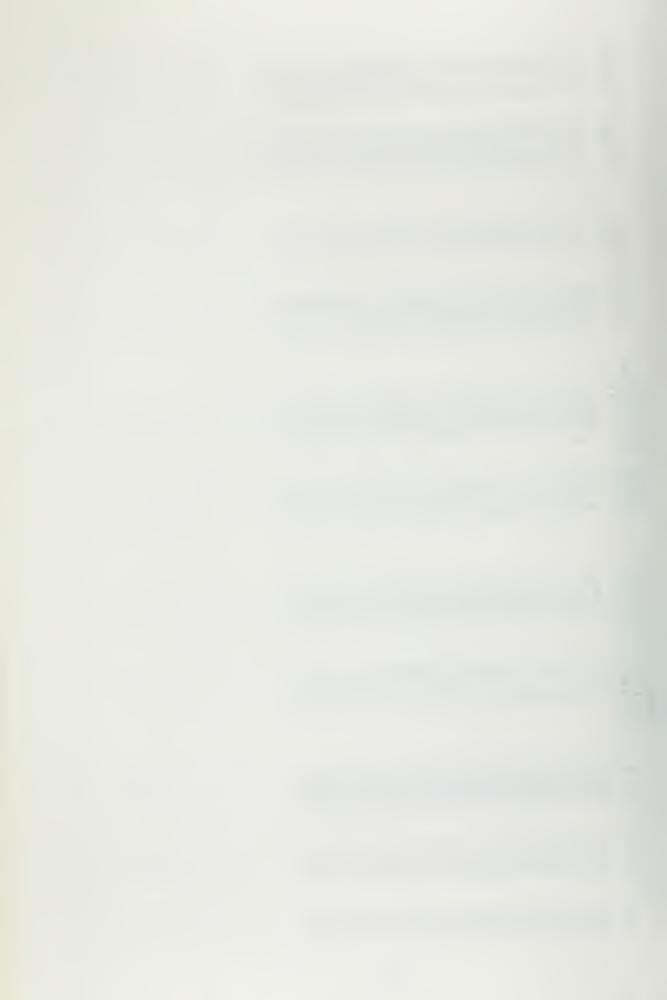
	0000		T ERROR DIST	1	
	. DS		PER CEN	00000000000000000000000000000000000000	
	00				
	= 71		RENCES		
	ΣΝ ΣΝ		DIFFE	000000000000000000000000000000000000000	
			Σ		
	00	0.0	ODIFIED ALGORIT DIST	2 2 2 1 1 1 1 1 2 2 2 2 2 2 2 2 2 2 2 2	
	DM1 =	VE = DTI =	NPS M BOEING TIME	1 1 3 3 3 3 11111111 1 1 1 1 1 1 1 1 1	
17	0.0073290	3.00	1-1C-1T-1 CS TABLES DIST	1 1 1 1 1 1 1 0 6 9 5 3 4 6 9 5 9 5 9 5 9 5 9 5 9 5 9 5 9 5 9 5 9	
FOR IDNO	DKG1 = DKG2 =	IREF =	NAVAIR O	1 1 3 3 3 3 3 3 4 4 4 4 4 4 4 4 4 4 4 4	
ICIENTS			ALT	6 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	
ON COEFF	M1 = 0.0		TAS	wwww4444nnnnwwwww4444nnnnwwwwww44 OONNOONNOONNOONNOONNOONNOO	
WEAP(CFOR	IBOTH	DEG		



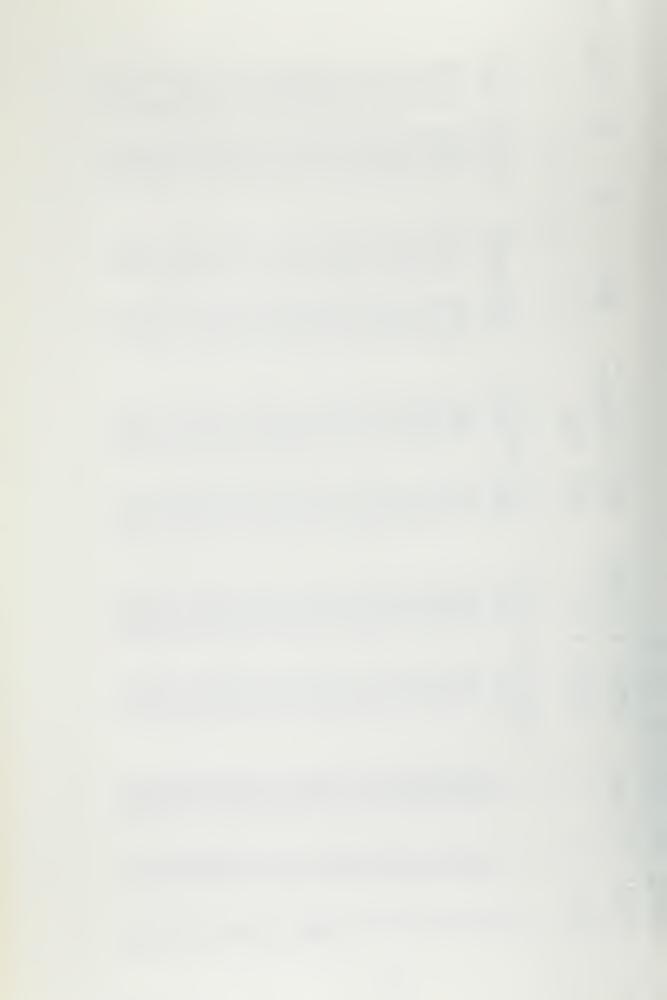
PER CENT ERROR TIME DIST	00000000000000000000000000000000000000	
RENCES		
DIFFE	70000000000000000000000000000000000000	
MODIFIED ALGORITHM DIST	1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	
NPS BOEING TIME	1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2	
1-1C-1T-1 CS TABLES DIST	1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	
NAVAIR O BALLISTI TIME	484049970404040404040404040404040404040404	
ALT		
TAS	44NNNNWWWWW4444NNNWWWWWWWWWWWWWWWWWA444NN NNOONNOONNOONNOONNOONNOONNOONNOONN	
DEG		



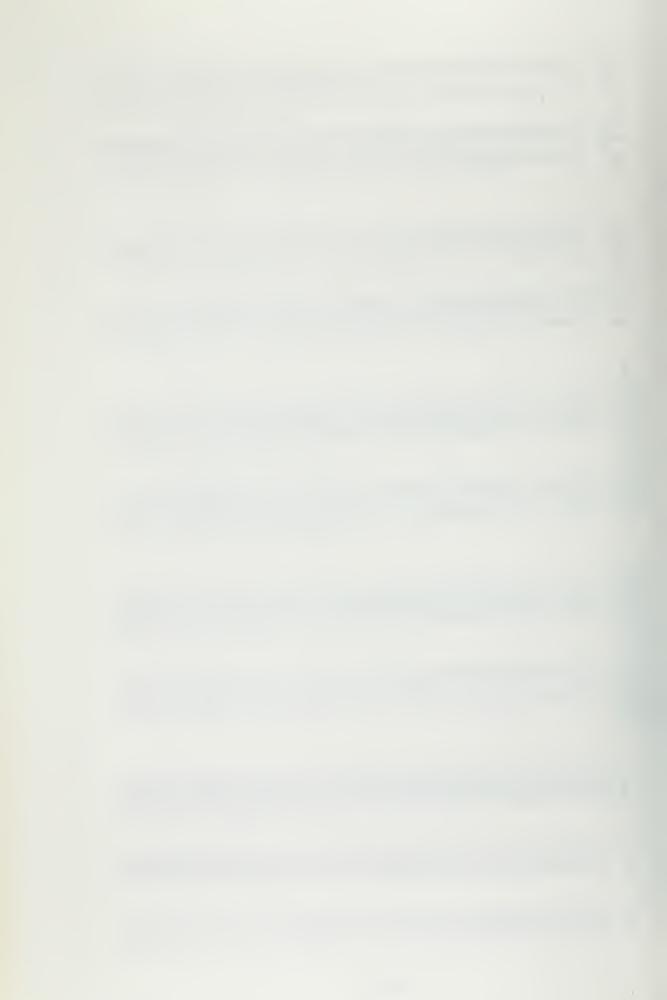
PER CENT ERROR TIME DIST	00000000000000000000000000000000000000
RENCES DIST	
DIFFE	000000000000000000000000000000000000000
MODIFIED ALGORITHM DIST	1 100000000000000000000000000000000000
NPS BOEING TIME	2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
1-1C-1T-1 CS. TABLES DIST	1140 8271114 10399439801430803 103901430803 84714305814430 847188886 847188886 847188886 847188886 8471888886 8471888886 847188888888888888888888888888888888888
NAVAIR OF	1
ALT	42222222222222222222222222222222222222
TAS	nnwwww.4444nnnnwww.444nnnn nnoonnoonnoonnoonnoonn 0000000000
DEG	



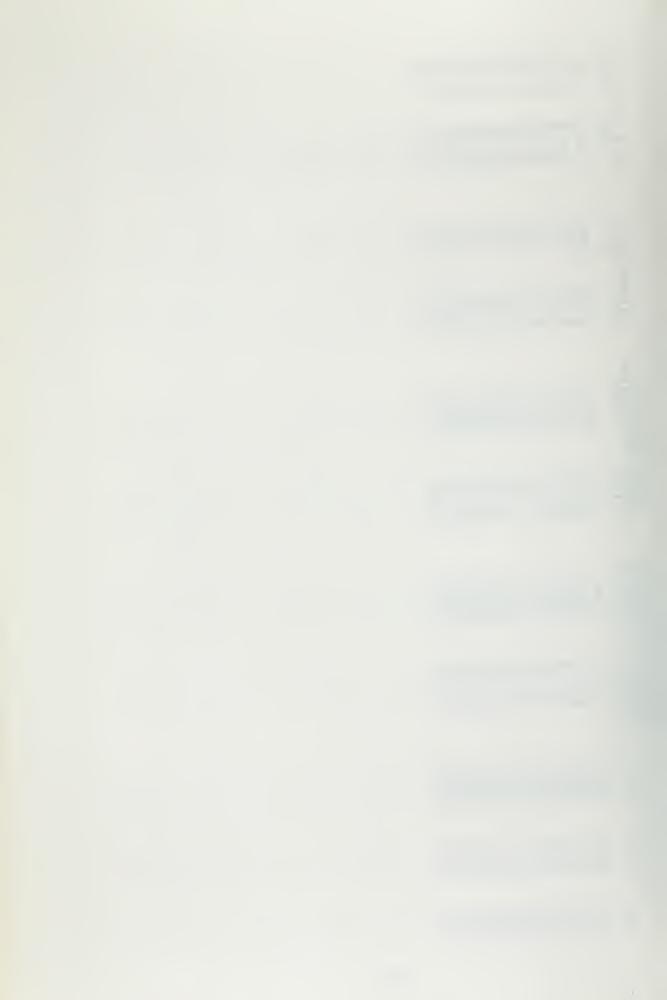
	0. DS = 0.6617000 0. SL =0002690		PER CENT ERROR TIME DIST	
WEAPON COEFFICIENTS FOR IDNO 18	VMUZ = FN =		DIFFERENCES TIME DIST	00000000000000000000000000000000000000
	$\begin{array}{cccccccccccccccccccccccccccccccccccc$	$VE = 0 \bullet 0$ $DTI = 2 \bullet 00$	NPS MODIFIED BOEING ALGORITHM TIME DIST	113.09.00 113.09.00 113.09.00 113.09.00 113.09.00 113.09.00 113.09.00 113.09.00 113.09.00 114.00 115
	$\begin{array}{rcl} DKG1 &= 0.0073290 \\ DKG2 &= 0.1716599 \end{array}$	IREF = 1 DMAX = 5.00	NAVAIR 01-1C-1T-1 BALLISTICS TABLES TIME DIST	
	M1 = 0.0 M2 = 0.0168950		ALT	11200000000000000000000000000000000000
		E = 1 H = 2	TAS	www.4444\n\n\n\w\w\w\w\w\\\\\\\\\\\\\\\\
	CFOR	ITYPE	DEG	00000000000000000000000000000000000000



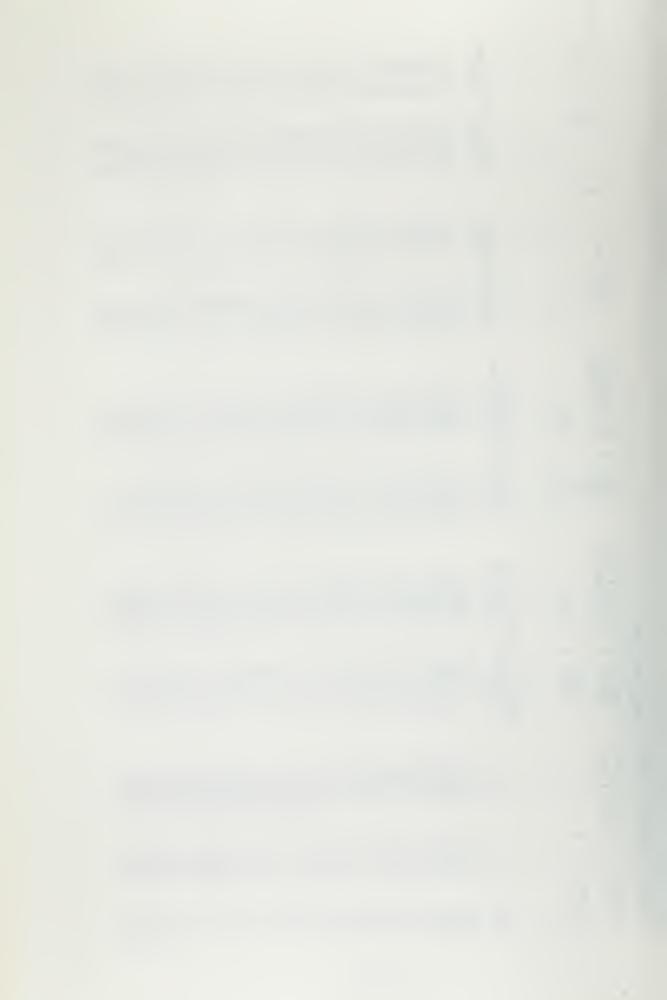
PER CENT ERROR TIME DIST	00000000000000000000000000000000000000
RENCES DIST	1
DIFFE	00000000000000000000000000000000000000
MODIFIED ALGORITHM DIST	を
NPS BOEING TIME	21/21/2
1-1C-1T-1 CS TABLES DIST	のいちにいいというに、
NAVAIR O	
ALT	
TAS	44NNNNWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWW
DEG	



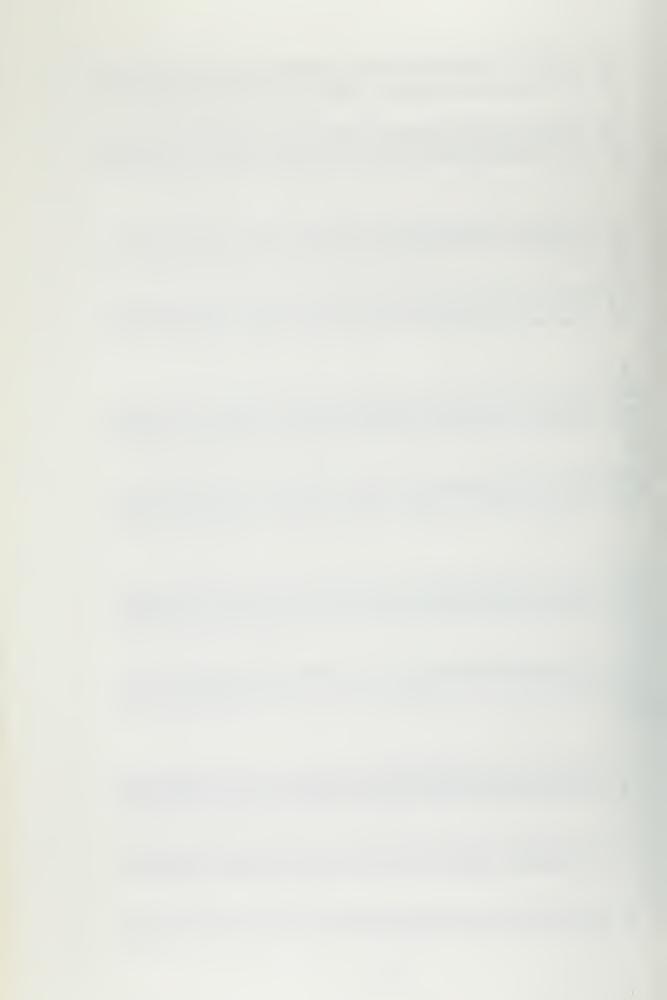
FRE	DIST	0	- I - 08	- ~	.+		_	m		m	\sim 1			~
ر	TIME	3.99	1.26	-0.33	0.34	-0.39	60.0	-0.48	0.45	-0.50	0.89	-0.47	1.87	-0.35
LL.		-108.	134	9	-9-	16.	m	31.	-1	45.	ا ا	52.	-31.	52.
1 FFF		0.58	00.00	-0.13	0.04	-0.16	0.01	-0.21	0.08	-0.23	0.18	-0.22	0.40	-0.17
MODIFIED ALGORITHM	DIS	4	3129	ð	ω	ð	m	ω.	9	2	4	∞	N	0
NPS FING	TIME	5.0	23.99	8	2.4	9.0	S. S.	3.2	6.9	S. 0	0.2	7.1	1.8	۳°
1-1C-1T-1 CS TABLES	DIST	S	3163.	8	49	07	72	25	86	40	04	53	15	64
NAVAIR O	TIME	4.4	23.69	8.1	2.4	0.8	5.5	ر ا	9.8	6.2	0.0	7.3	1.4	8.4
	ALT	4500.	0000		4000	2	ഗ	m	S	4	Q	4	_	15000.
	TAS	S	200	0	S	S	0	0	S	S	0	0	S	2
	DEG	4	145	9	9	9	Ġ	9	9	9	9	9	9	9



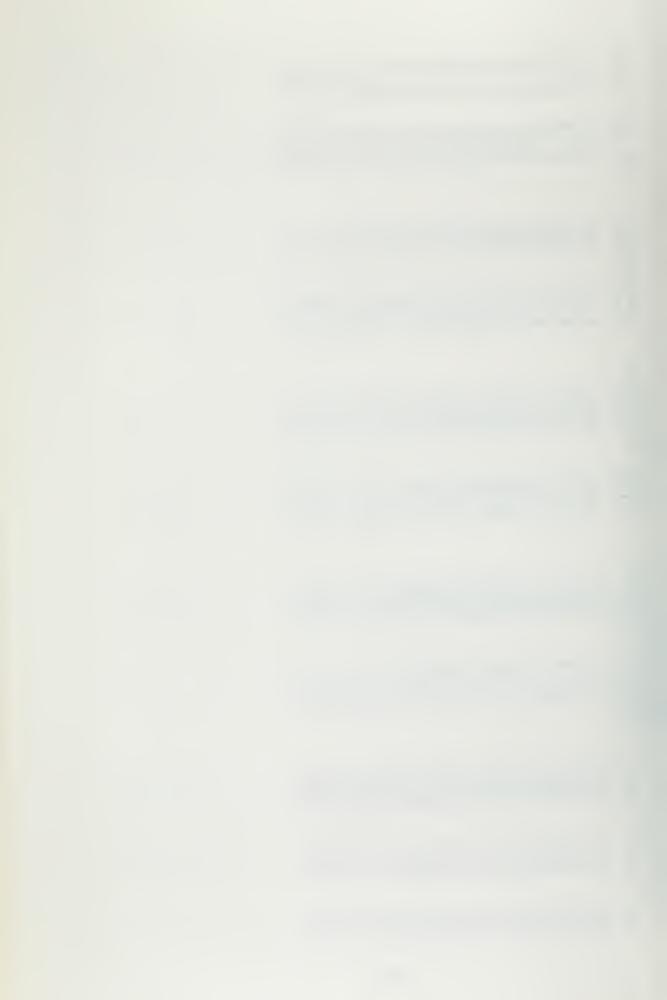
FOR WAR	COEF = 2.	FICIENT S 2572994 0111360	FOR IDNO 2 DKG1 = DKG2 =	0 0.0081750 0.1688499	ZX I	• •	= Z N N H	0 • 0 SL = 4
I 1 7 7 P E I B O 1 H			DMAX = DMAX = NAVAIR 01-	1 5.00 -1C-1T-1 S TABLES	N D S E E I E I E I E I E I E I E I E I E I	0.0 2.00 0DIFIED ALGORITHM	H C N	C F F
ш 00000	A 00000 00000	ALT 000000000000000000000000000000000000	71ME 9.87 21.52 10.67 22.24	0157 3687 5106. 5741.	7 I M M M M M M M M M M M M M M M M M M	0 1 S	I ME DI 03 1 02 02 02 02 02 04	11ME D -0.27 -0 -0.20 -0
			400004 600000	າພດຜບາພ,	100mour	36277 934	111 - 100 -	00.00 00
000000		00000000000000000000000000000000000000	1400000 1500000	18540 1965 1965 1965 1965 1965 1965 1965 1965	1400000 19140 19140	2004074 00000000		00000000000000000000000000000000000000
		0000000	180700000000000000000000000000000000000	10044000 1007mn	0000000	884709i 3012i306v	00000000000000000000000000000000000000	00000000000000000000000000000000000000
000000	vwwww44 voovvoo	255000 255000 255000 35000 35000	60.50 1.8.50 1.8.32 1.8.33 1.0.05 1.0.05	441284 471284 471288 60158 500	60.87 138.522 158.37 17.118	997888 941408 941409 969146 56974	0.37 0.002 0.004 0.00 0.008 0.008 0.008 0.008 0.008 0.008 0.008	-00.024 -00.024 -00.020 -00.038 -10.000



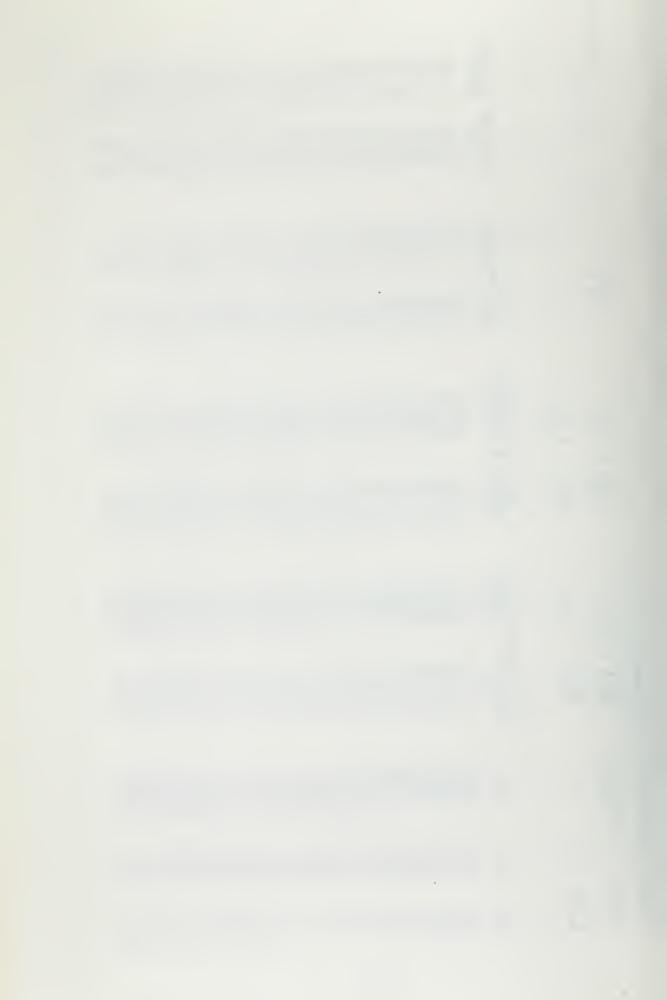
NT ERROR DIST	1111111
PER CE	00000000000000000000000000000000000000
RENCES	
DIFFE	00000000000000000000000000000000000000
¥	
ODIFIED ALGORITI DIST	$\begin{array}{c} 4040 \text{L} \text{W} + \text{W} +$
NPS M BOEING TIME	2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
S TABLES DIST	50000000000000000000000000000000000000
01 1C	.0.000 +10.000 000 +00.1 +010-1010 +00.0 0 -1.01010 +00.0 1
NAVAIR BALLIST TIME	-0-80084-00-800-10-10-10-10-10-10-10-10-10-10-10-10-1
ALT	
TAS	44NNNNWWWWWA444NNNNWWWWW4444NNNNWWWWWWA444NN NNOONNOONNOONNOONNOONNOONNOONNOO
DEG	11111111111111111111111111111111111111



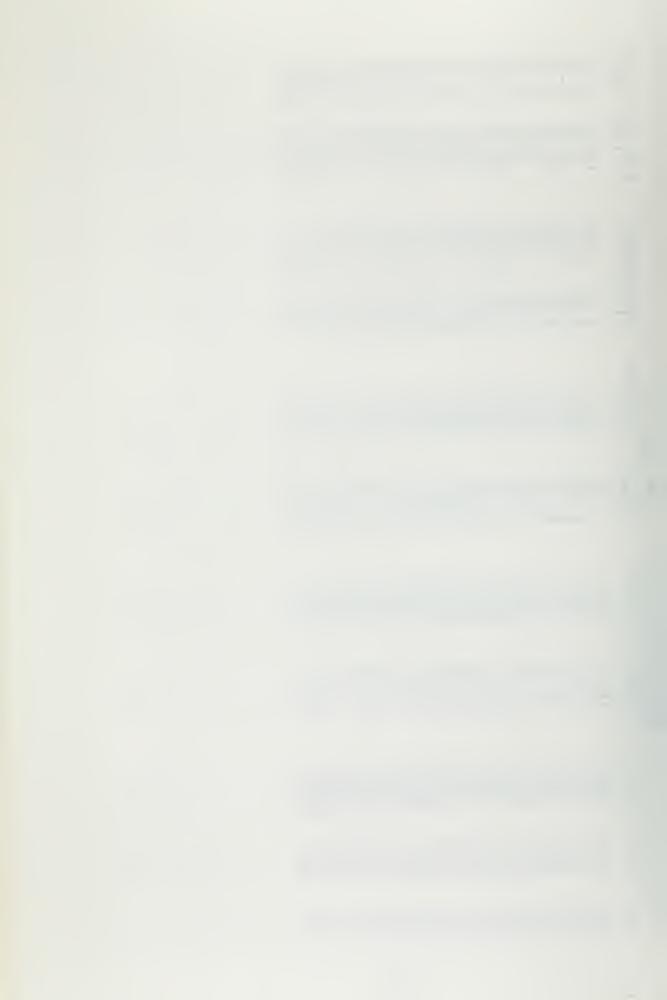
NT ERROR DIST	00000010700000000000000000000000000000	ω
PER CENTIME	1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	4.
RENCES DIST	0 101w0wv1010001110101010	36.
DIFFE	00000000000000000000000000000000000000	
MODIFIED ALGORITHM DIST	を できて できて できて できて できて できて できて でき と と と の の か と と と の の か と と と の の か と と と の の か と と と の の か と と と の の か と と と と	03
NPS N BOEING TIME	W Z Z Z W W W 4 4-14 4-14-100-100-100-100-100-100-100-100-100	90
1-1C-1T-1 CS TABLES DIST	######################################	90
NAVAIR O BALLISTI	2 2 2 2 2 3 4 4 4 4 4 6 6 6 6 6 6 6 6 6 6 6 6 6 6	9.6
ALT	40wrwawa404114141411000000000000000000000000	5000
TAS	NNWWWWW4444NNNNWWWWW4444NNN NNOONNOONNOONNOON 00000000000000	500
DEG	00000000000000000000000000000000000000	0

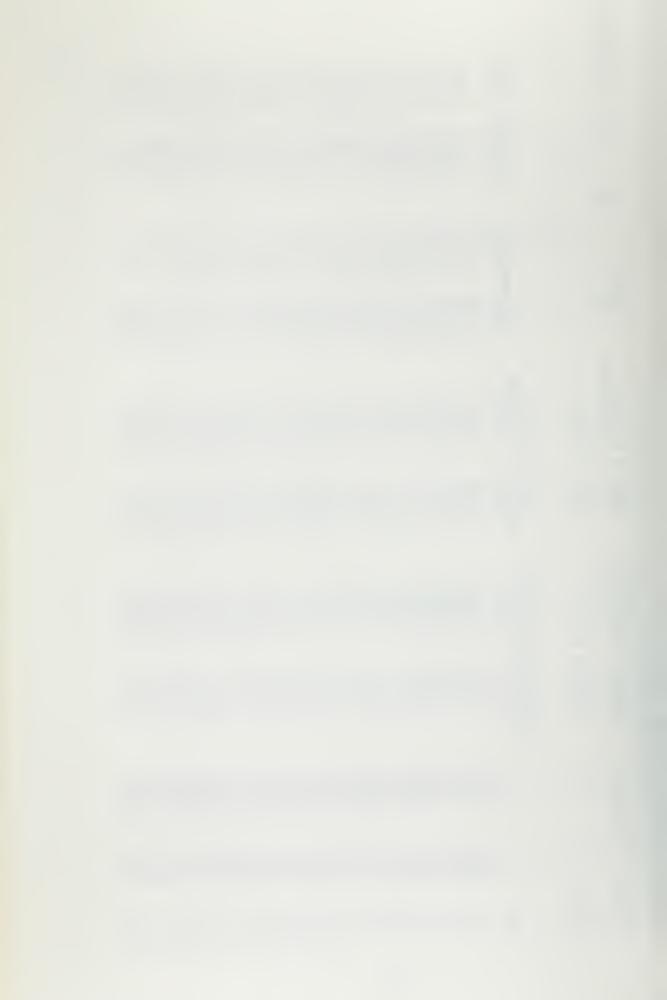


WEAPON	COEFF	ICIENTS	FOR IDNO 2	~						
CFORM1 CFORM2	= 2.2 = 0.1	403994 178000	DKG1 = DKG2 =	00.0	DM1 = DM2 =	0.0	KMUZ FN =	11	00.	S = 4.00000000
ITYPE IBOTH	== 5		IREF DMAX =	5.00	VE = OTI =	0.0 1.62				
DEG	TAS	ALT	NAVAIR 01- BALLISTIC TIME	-1C-1T-1 S TABLES DIST	NPS MO BOEING A	DIFIED LGORITHM DIST	DIFFERE TIME	ENCES DIST	PER CE	NT ERROR DIST
000000000000000000000000000000000000000	444NNNN4444NNNNA444ANN OONNOONNOONNOONNOONNOO OOOOOOOOOO	w w w willing www.wx4x4w4www. wowowowooooooooooooooooooooooooooo	21212121211111212121212121212121212121	40000000000000000000000000000000000000	2121220211111414136を122222222222222222222222222222222222	00000000000000000000000000000000000000	00000000000000000000000000000000000000		01010001101101101000010000000000000000	



	PER CENT ERROR TIME DIST	11	.81 -6.3 .74 -8.6
	DIFFERENCES TIME DIST	1.21 -0.888 -0.5822 -0.8882 -0.8822 -1.309 -1.309 -1.100 -1.10	.51 -25 .09 -41
STREETER SO	NPS MODIFIED EING ALGORITHM IME DIST	86 3 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2	04 377 42 442
	S TABLES BOILD DIST	494448000040444444444444444444444444444	031. 13 839. 38
	NAVAIR 01- BALLISTIC TIME	8 31912 2121212 212120 8 31912 2121212 212130 8 69200122 6920 1 19140 4 1914 6920 6920 6920 6920 6920 6920 6920 6920	5.30
	ALT	10000000000000000000000000000000000000	000
	TAS	NN4444NNNN4444ANNNN4444ANN NNOONNOONNOONNOONNOO	50
	DEG		450





IT ERROR DIST	
PER CEN TIME	1041L20100100000000000000000000000000000
RENCES	11111 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
DIFFE TIME	00000000000000000000000000000000000000
_	
ODIFIED ALGORITHM DIST	
NPS M BOEING TIME	21122111121111111111111111111111111111
STABLES DIST	00000000000000000000000000000000000000
100	
NAVAIR BALLIST TIME	######################################
ALT	-ин-и-ш-и-шишишишишичичта тачи-ш-и-ш-и-ш-и-шишишишишишишишишишишишиш
TAS	44NNNNWWWWW4444NNNNWWWWW4444NN NNOONNOONNOONNOONNOONNOONNOONNOO
DEG	



IT ERROR DIST	-4.89 -3.08 0.46	o ry c	101	SO.	450.	40
PER CEN TIME	3.99 2.12 -0.35	S	2-1-0	250	ω	00
RENČES . DIST	-117. -79.	13.	74m 74m	w 10.0	. n.	13.
DIFFER	0.68	-0. -0. -0. -0. -0.	-0.03 -0.24	0.05	-0.16 -0.16	-0.00
MODIFIED ALGORITHM DIST	2279. 2483. 1257.	4 6 7 u	して	286	100	0.5
NPS M BOEING TIME	17.66 23.98 15.14	44r	-88 -92	0.00	***	200
-1C-1T-1 S TABLES DIST	2396. 2562. 1251.	യസ്	りつす	1	η ω	4
NAVAIR OL BALLISTIC TIME	16.98 23.48 15.19	441	9.3	9.0	2.4.0 2.4.0	20.0
ALT	4500. 6000.	000		000	200	200
TAS	000 000	OW	200	S	001	S
DEG	14450	99	9	99	99.	



APPENDIX C

This appendix is a listing of the output from the cockpit of various A-6E aircraft recorded at the instant the weapon was released.

The following is a brief description of the parameter headings as they appear on the listing.

TAS = true airspeed in knots

TH = true heading in degrees relative to true north

WDIR = wind direction in degrees true

WKTS = wind speed in knots

GT = ground track in degrees true

GS = ground speed in knots

RA = release angle in degrees

VZ = vertical velocity in knots x 10

VSEP = vertical separation in feet

DA = depression angle (search radar) in degrees

SR = slant range to target in feet

TOF = time of fall in seconds

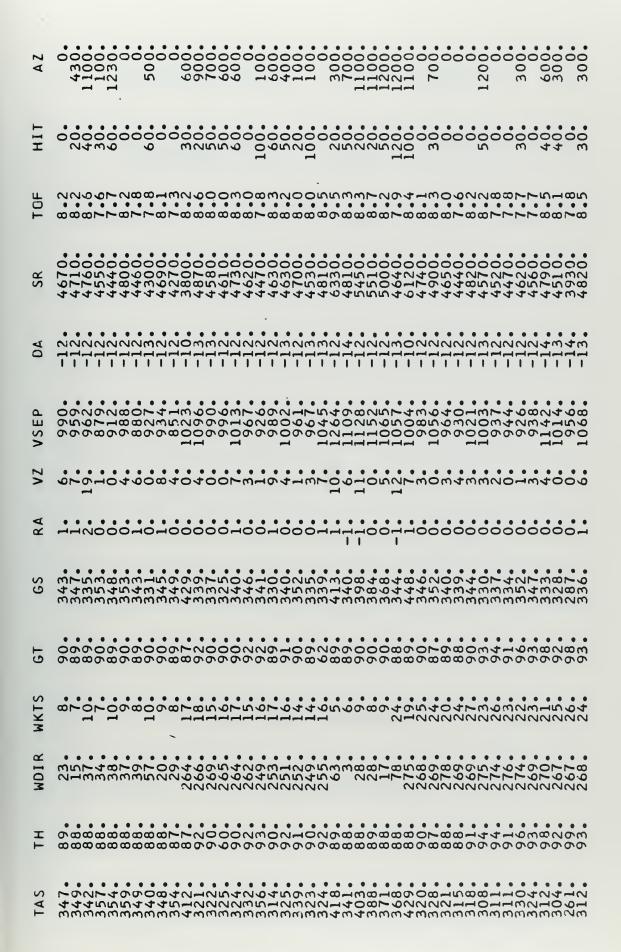
HIT = his distance from target in feet

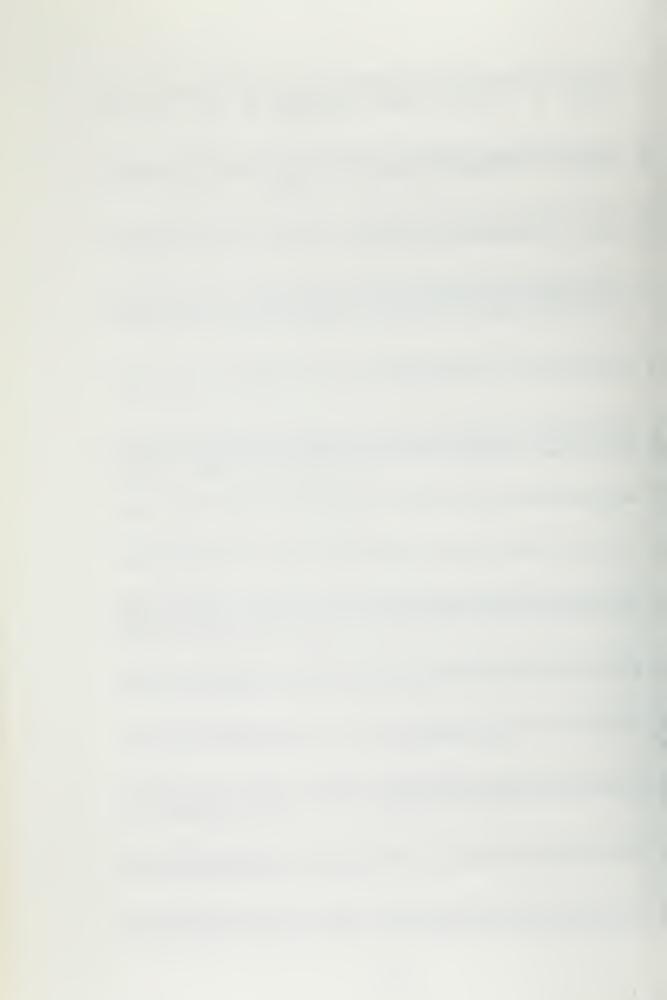
AZ = hit azimuth in clock code

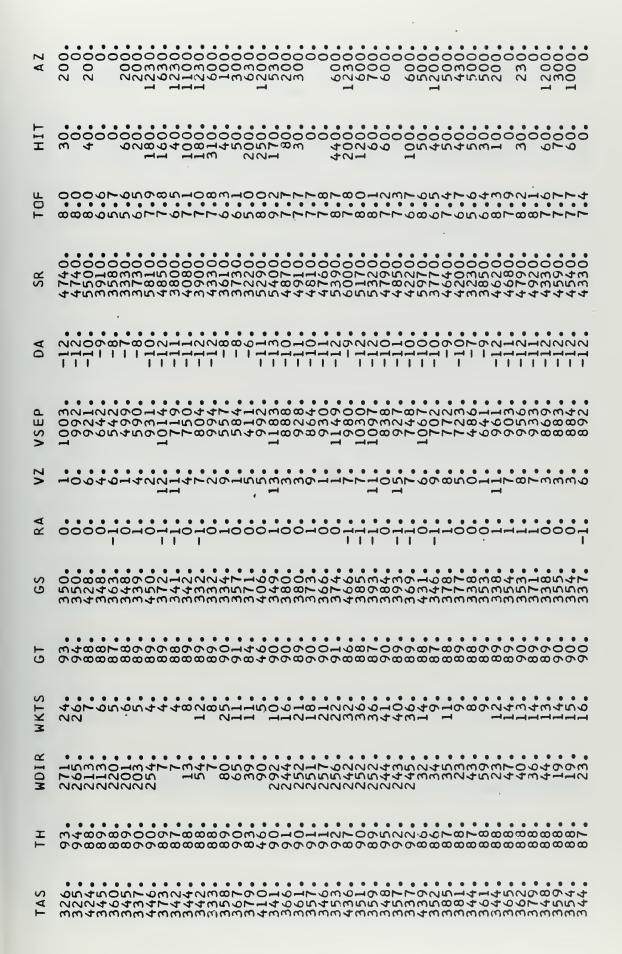


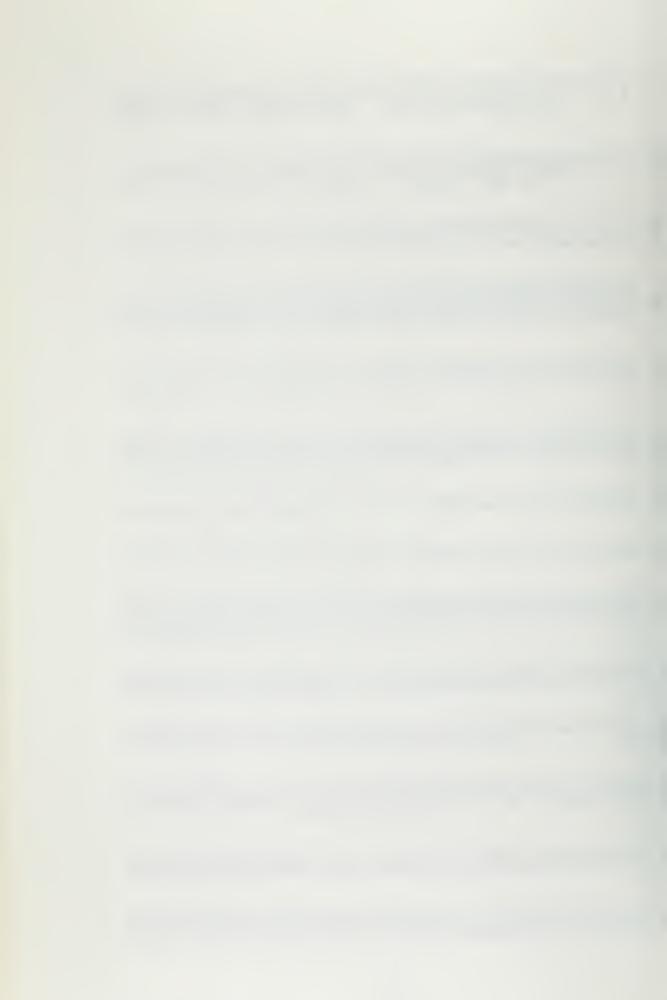


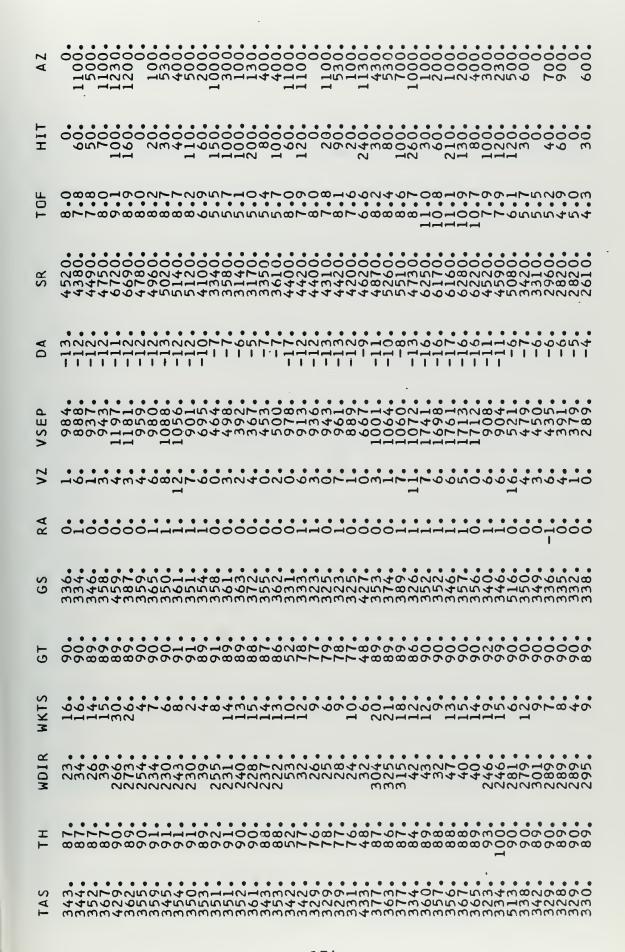


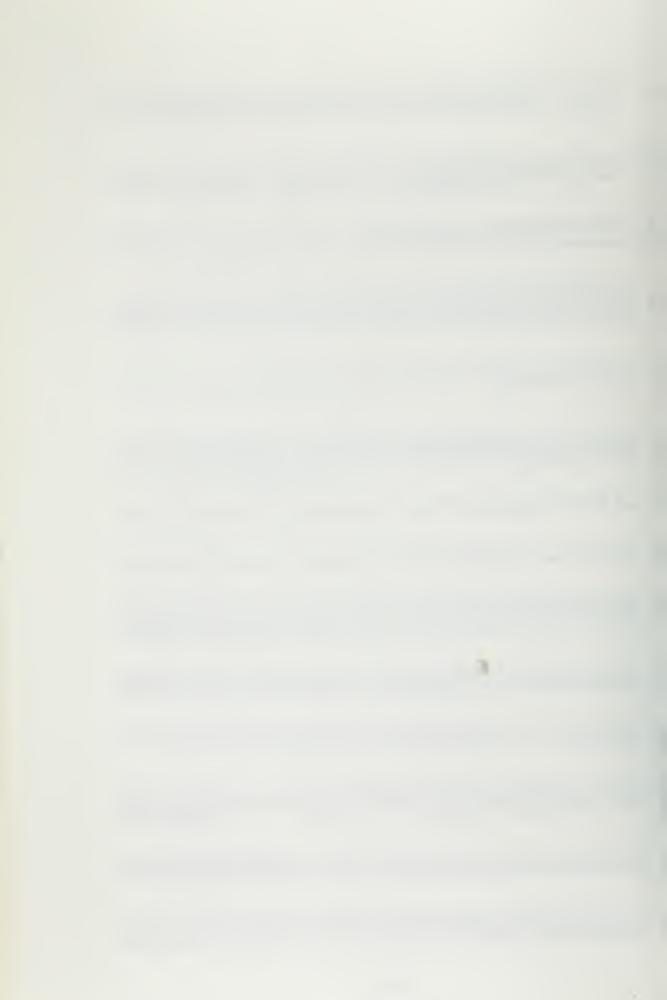




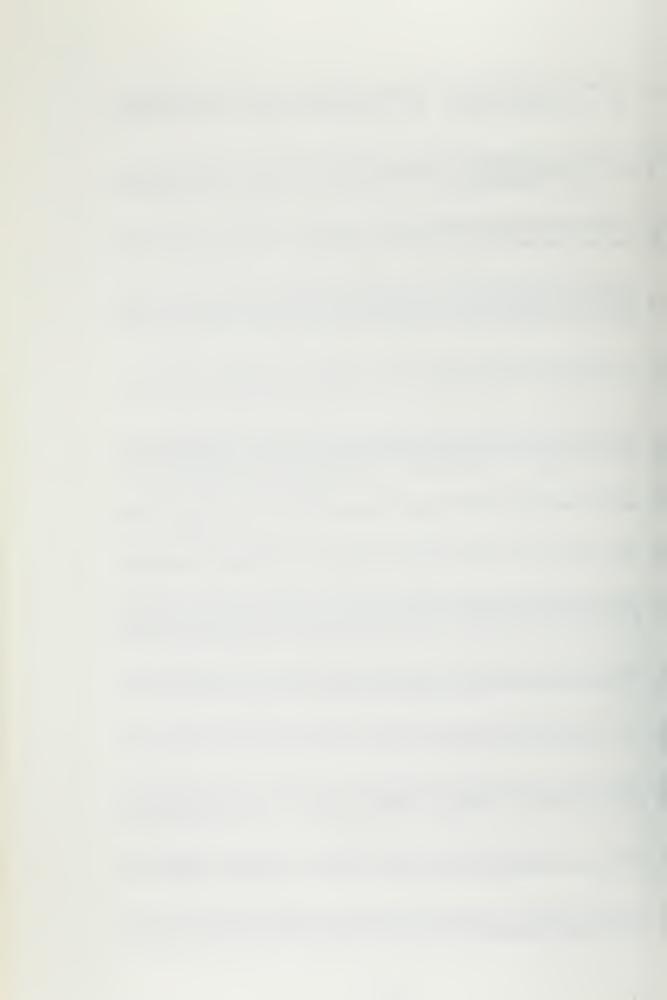




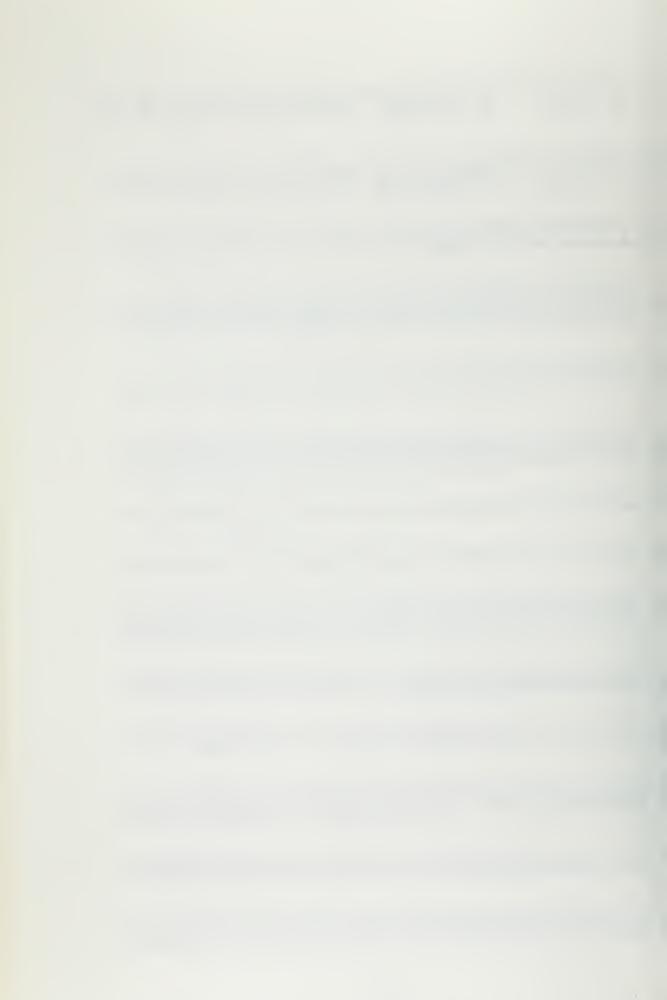


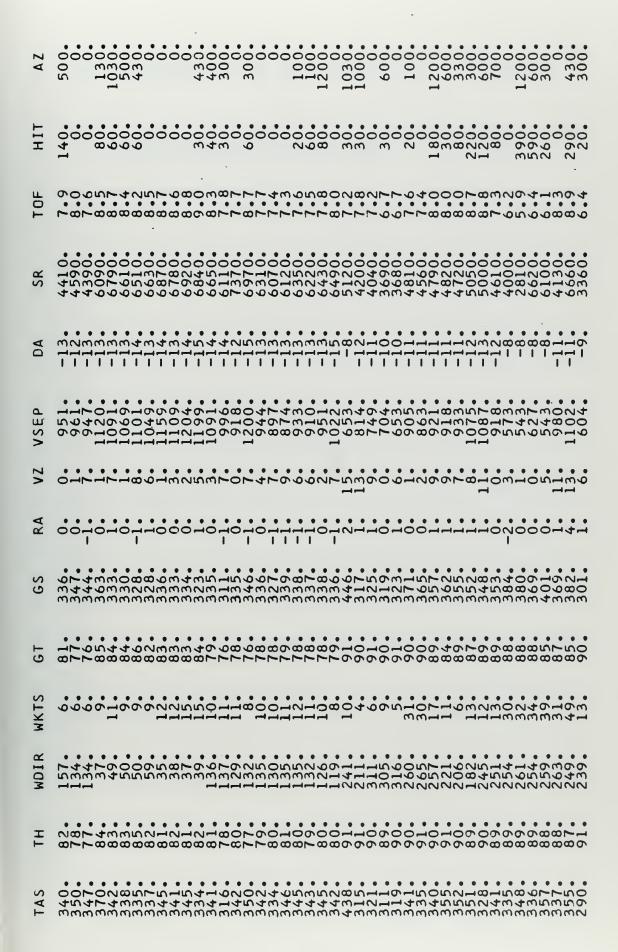


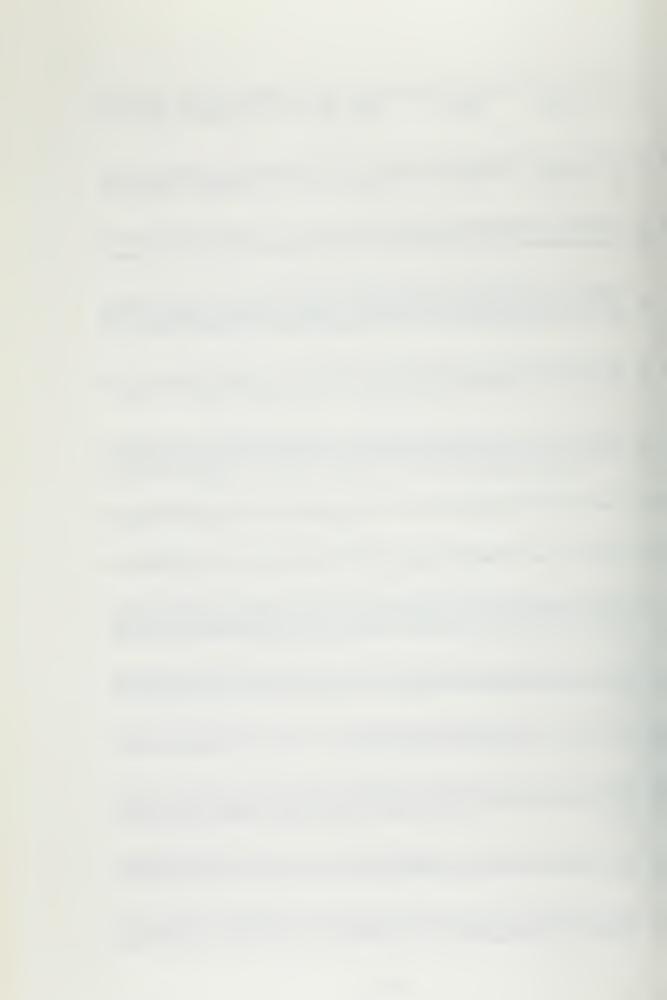


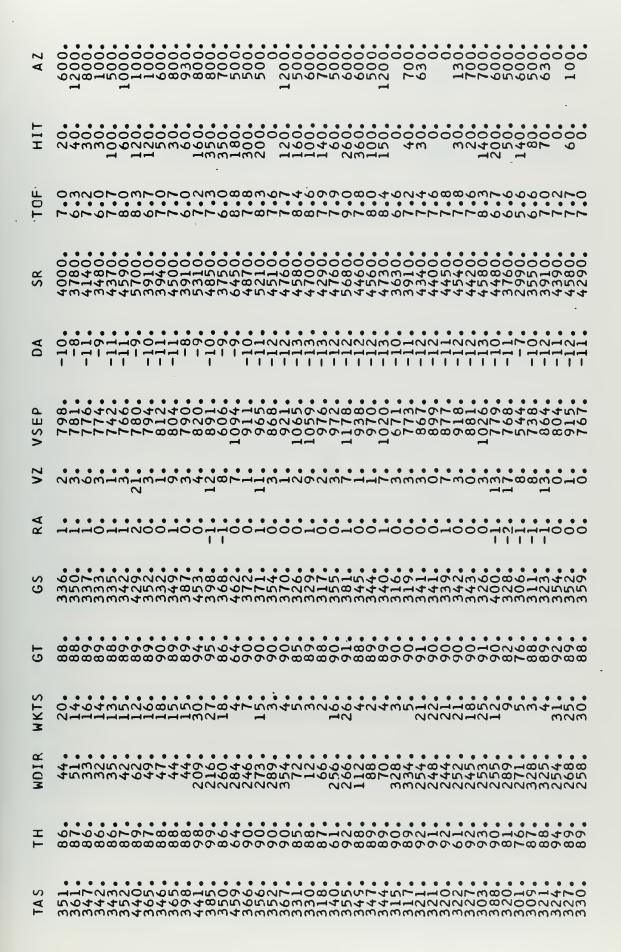


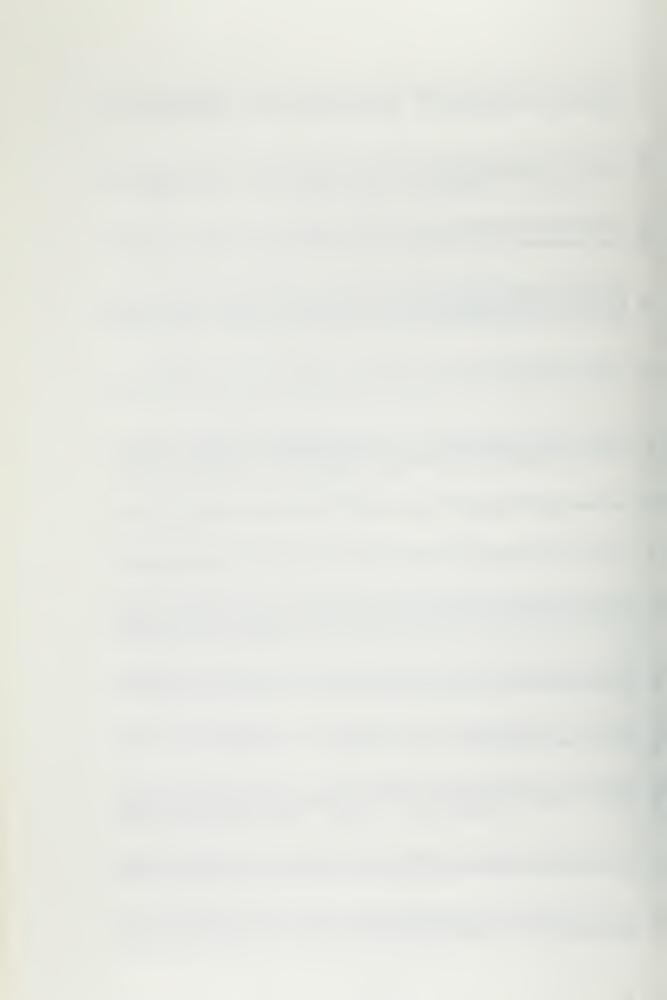


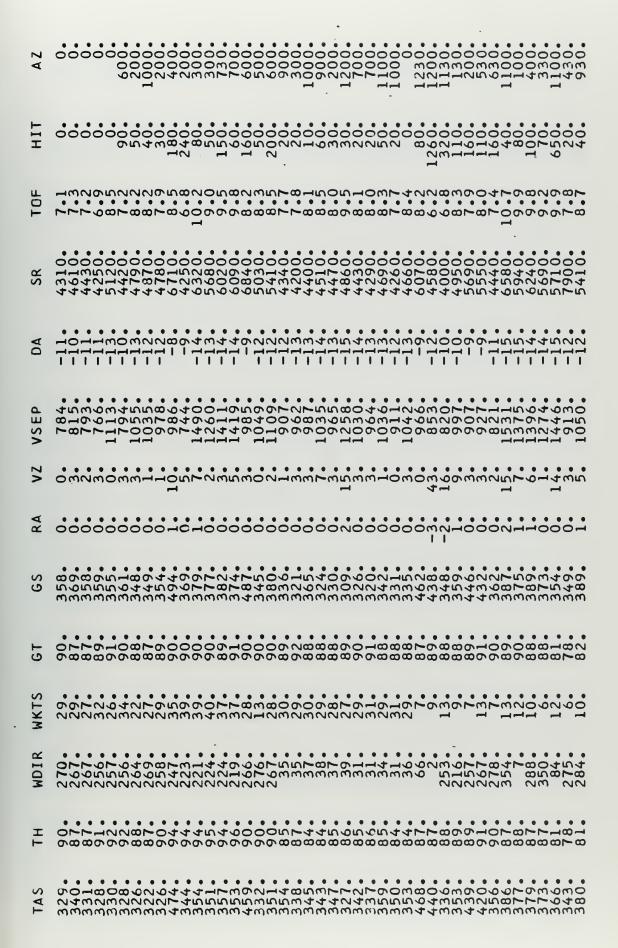


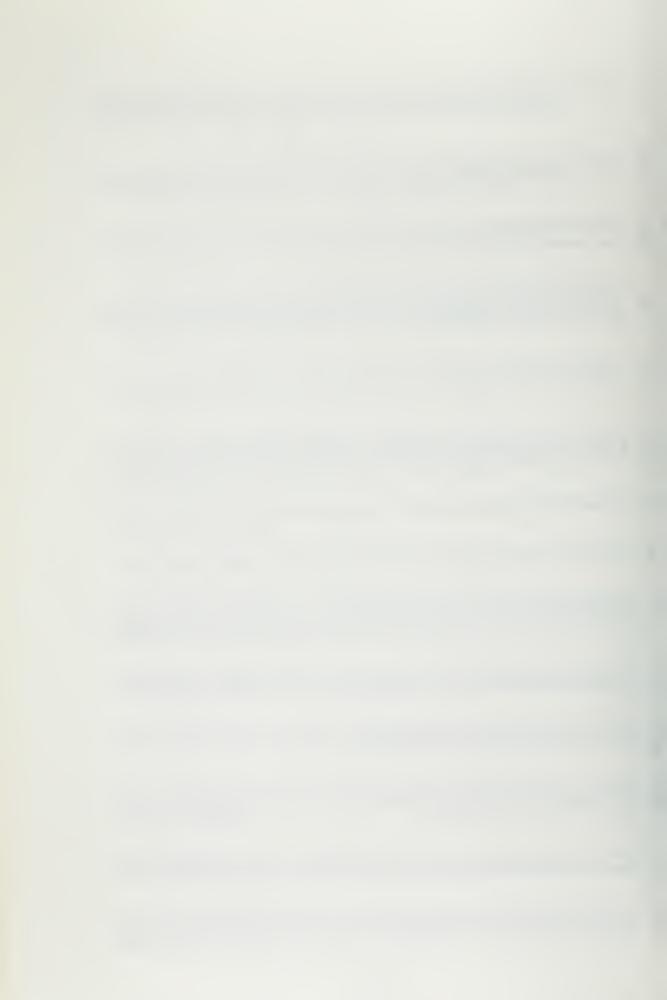




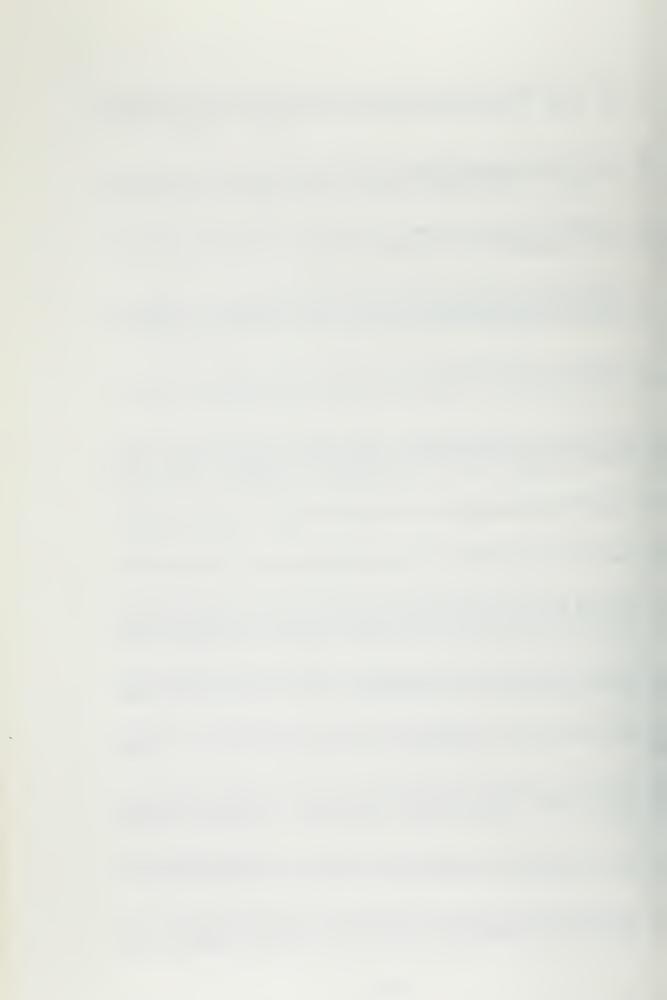


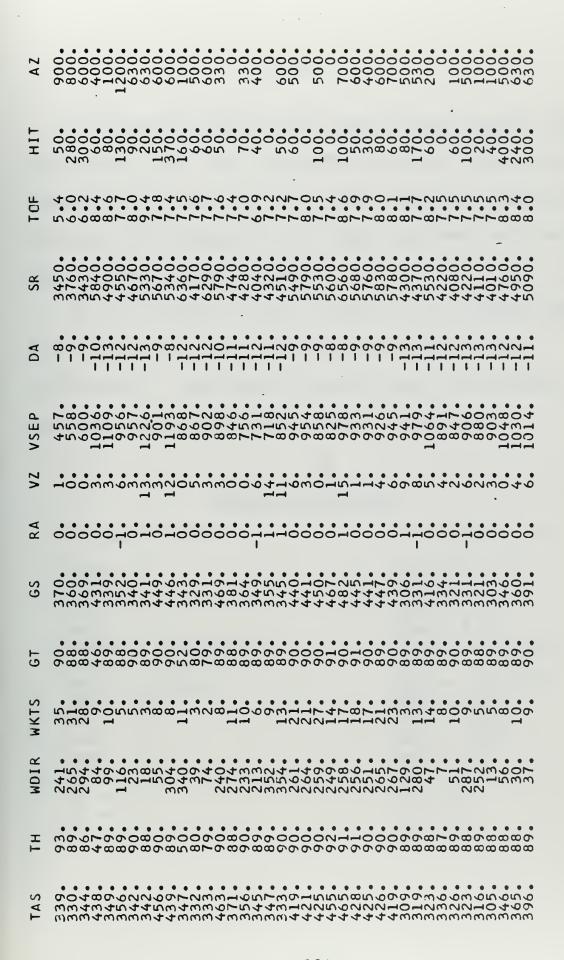


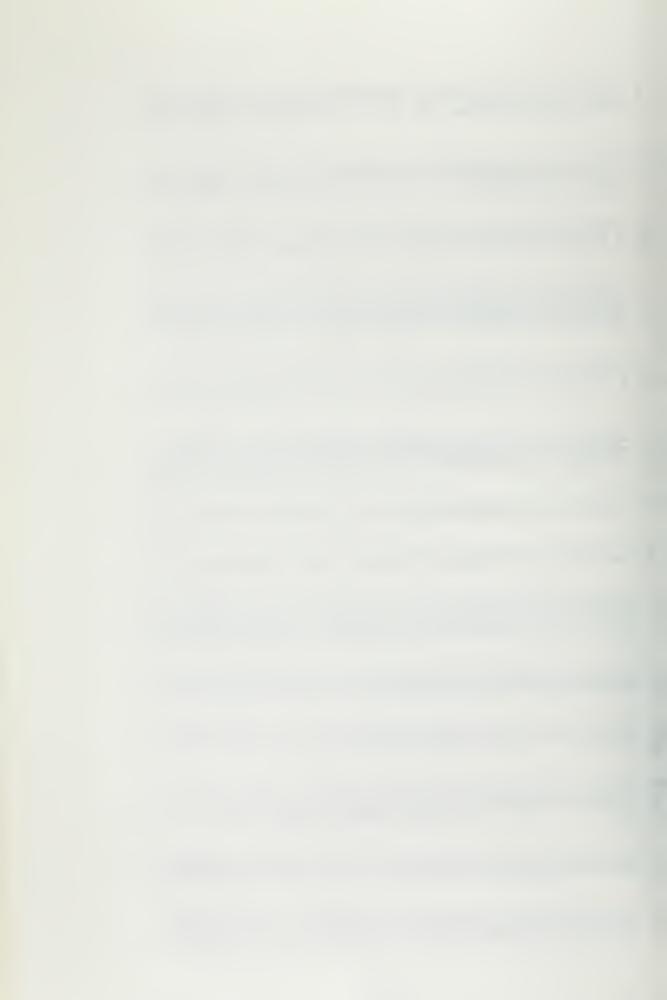












APPENDIX D

This appendix contains the output which compares the experimental data with the FORTRAN version of the ballistics algorithm using the old set of mach, drag, and weapon coefficients. The weapon used for this experiment was the MK-76 MOD-5 25 pound practice bomb.

The coefficients are all assigned in the DECODE subroutine (see Ref. 3 for further explanation of these
coefficients). Both sets of coefficients, old and new,
are summarized here for the reader's convenience.

Weapon Coefficients

	O	\mathbf{I}_{C}	i C	0	ei	ΞĒ	ic	ĺ	en	ts
--	---	------------------	-----	---	----	----	----	---	----	----

IREF = 2
IBOTH = 1
ITYPE = -1
DMAX = 3.0
CFORM1 = 0.0039077
CFORM2 = 0.0
DKG1 = 0.0063648
DKG2 = 0.0
DM1 = 0.0

VMUZ = 0.0 FN = 0.0 VE = 0.0 SL = 0.0 DS = 0.0

DM2 = 0.0

New Coefficients

IREF = 2 IBOTH = 1 ITYPE = -1 DMAX = 6.0 CFORM1 = .1064453 CFORM2 = 0.0 DKG1 = -.0043918 DKG2 = 0.0 DM1 = -.270 DM2 = -.270 VMUZ = 0.0 FN = 0.0 VE = 0.0 SL = 0.0 DS = 0.0



New Coefficients

01d Coefficients

MSTG=1 Go to (32,33,34,51), IREF CC(1,1,MSTG) = 1.572924E-3 CC(1,2,MSTG) = 0.0 CC(2,1,MSTG) = 4.67840889E-2 CC(2,1,MSTG) = 4.67840889E-2 CC(2,3,MSTG) =109711069 CC(2,3,MSTG) =116380157 CC(3,1,MSTG) =116380157 CC(3,2,MSTG) = -9.76706845E-2 CC(3,3,MSTG) = -9.76706845E-2 CT(1,MSTG) = .834 CT(2,MSTG) = .834 CT(2,MSTG) = .977 If (IBOTH.EQ.1) go to 51	CC(1,1,MSTG) = .173244 CC(1,2,MSTG) = 0. CC(1,3,MSTG) = 0. CC(2,1,MSTG) = .215467 CC(2,2,MSTG) = .285067 CC(2,3,MSTG) = .489778 CC(3,1,MSTG) = .0039111 CC(3,2,MSTG) = .5880 CC(3,2,MSTG) = .5880 CC(3,2,MSTG) = .5880 CC(3,2,MSTG) = .5880 CC(3,2,MSTG) = .5880 CT(2,MSTG) = .27 CT(2,MSTG) = .27 CT(2,MSTG) = .52 If (IBOTH.EQ.1) go to 51 If(IREF.EQ.1) go to 51
311	ee e
	•
31. Go to (32,33,34,51), IREF CC(1,1,1) = 1.572924-03 CC(1,2,1) = 0.0 CC(1,3,1) = 0.0 CC(2,1,1) = 4.678409E-02 CC(2,2,1) = 0.109711069 CC(2,3,1) = 6.654801E-02 CC(3,1,1) = 0.116380157 CC(3,1,1) = 0.217643894 CC(3,2,1) = 0.217643894 CC(3,3,1) = 0.217643894 CC(3,3,1) = 0.217643894 CC(3,3,1) = 0.217643894 CC(3,1,1) = 0.977 If (IBOTH-1) 33,51,33	33 CC(1,1,1BOTH) = 3.53503924 CC(1,2,1BOTH) = -3.34778216 CC(1,3,1BOTH) = 2.87262413 CC(2,1,1BOTH) = 11.2616503 CC(2,2,1BOTH) = -27.4162512 CC(2,3,1BOTH) = -27.4162512 CC(3,1,1BOTH) = -23.7915472 CC(3,2,1BOTH) = -23.7915472 CC(3,2,1BOTH) = -44.2607764 CC(3,3,1BOTH) = -14.4996046 CT(1,1BOTH) = 0.622 CT(2,1BOTH) = 0.885 Go to 51
ന ന -	n



.230347

.167644

-.194037 .401478

CC(1,1,MSTG) CC(1,2,MSTG) CC(1,3,MSTG) CC(2,1,MSTG) CC(2,2,MSTG) CC(2,3,MSTG) CC(3,1,MSTG)

CC(3,2,MSTG)

-2.03275E-02 = 2.44682E-03

7.33246E-02

= -0.164612

CC(1,3,1) CC(2,1,1) CC(2,2,1) CC(2,3,1) CC(3,1,1) CC(3,2,1) CC(3,2,1)

= 0.401478

-0.194037

-0.230347 = 0.104115

34

= 0.167644

-2.03275E-2 2.44682E-3

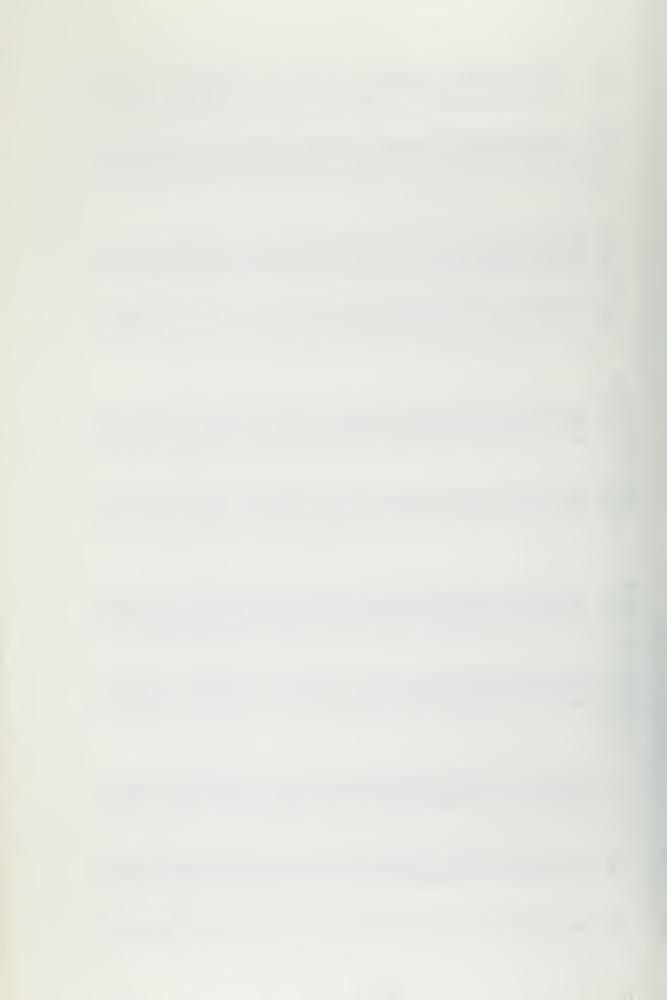
7.33246E-2 -.164612

> CT(1,MSTG) =CC(3,3,MSTG)

CT(2, MSTG)

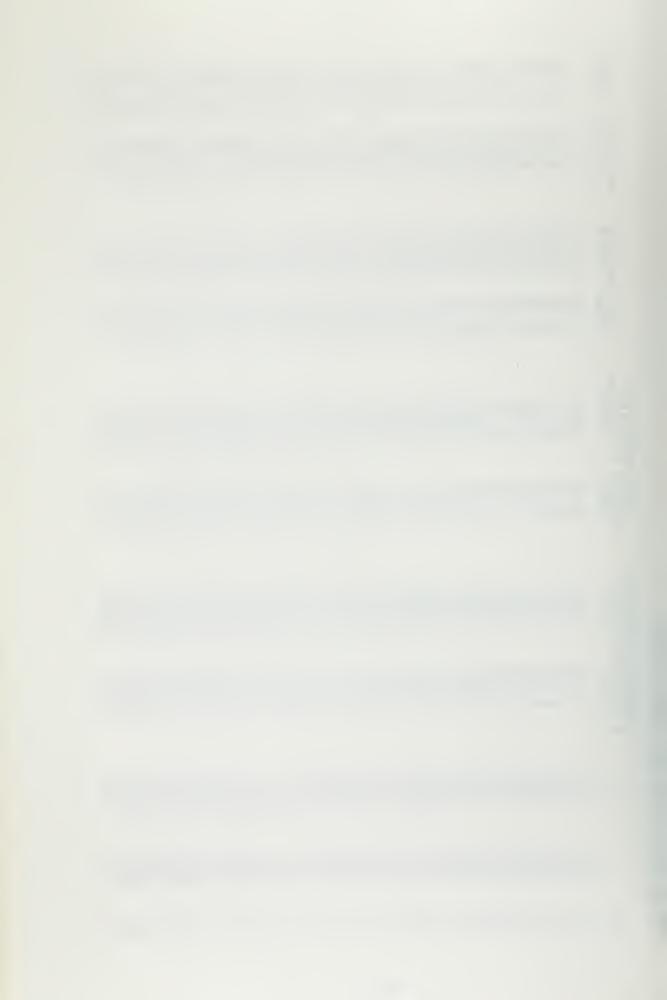


ROR ER OI SC EN EN EN EN EN くののちんのをしたらしてというとしてもあるとのものもらりとしるりとのちんとしてもらってもらってもらってもらってもらってもっているというとしてもっているというというというというというというというというといい ER T ENCES ERE DIFFE TIME \$\text{Purally and purally and part of the purally and part of the purally and part of the purally and part of



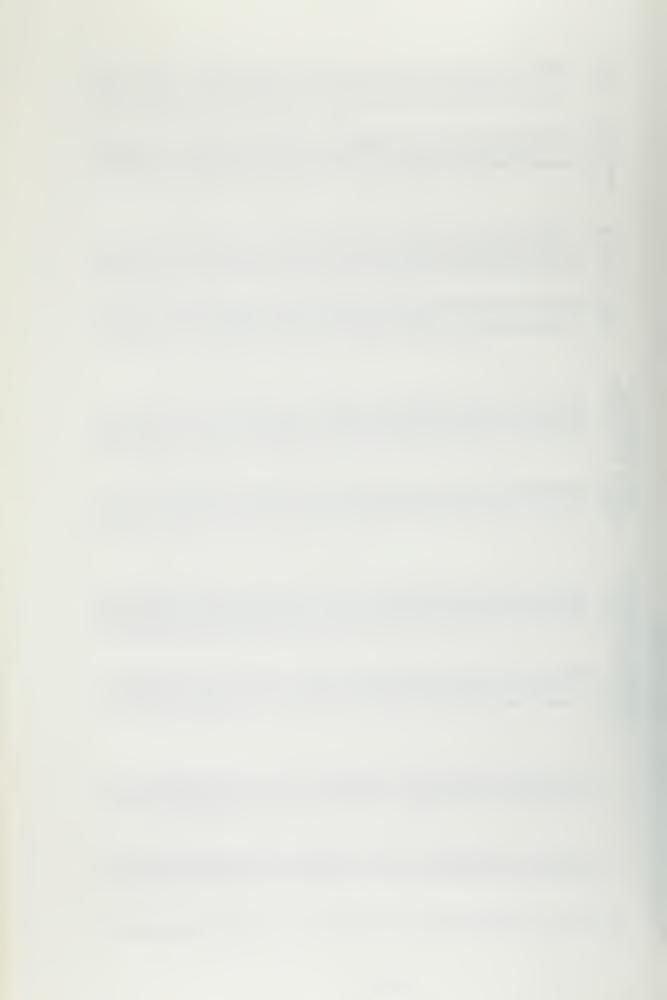
10
2
5
-
<u></u>
$\overline{}$
0
COEFFICIENTS
-
-
9
Ų
O
_
OLD
=
U
_
土
MITH
=
-5
MOD-5
=
2
_
-76
-
J

CENTER	TIME DIST	1.20 -4.75 1.20 -4.38 0.69 -5.11	.11 -2.7 .59 -4.3	.24 -5.1 .25 -7.6	.32 -4.4 .83 -7.0	.21 -4.9 .21 -4.9	07 - 20.0	. 36 . 36 . 64 . 64 . 6 . 6 . 6 . 6 . 6 . 6 . 6	. 52 . 85 - 20 . 6	62 -4.9	0.305.3	.31 -5.1	0.68 -5.9 0.82 -19.0	.72 -4.6 .84 -7.8	0.14 -5.9	.11 -3.2 .42 -6.6	.80 -3.7 .10 -6.8	.20 -4./ .85 -7.3	38 -6.6	1.37 -5.7 0.16 -6.7 0.54 -6.1	
ERENCES	DIS	-245. -222. -219.	129	315	318	130	13	26	23	1. 2.1 3.1	11	22	724	27	23	200	267	357	25	272	
DIFF	IME	0.10 0.10 0.06	• 0	00	0-	100	• • • • • • • • • • • • • • • • • • •) · · ·	00.	000	00	• 0	00	00	00		0.	00,	10.	000	
MODIFIED ALGORITHM	DIST	51:59. 5070. 4274.	447	44. 44.	986	000	200 200	14 0 0 0	32,	4 m	247	30	17	33	101	233	337	239	414	1140	
NPS	TIME	8.10 8.60 8.06	621	• • •	~ ŏ•	٠.٠	• • • • • • • • • • • • • • • • • • •	74.	00	401	-2	06.	وين	52	.7	w Or	0.0	ייסני	`(⊃ & &	
DEL IVERY EZE DATA	DIST	5404. 5292. 4492.	61 61 61	46	800 401	196	, w	74	45°	500	55	52	450	2089	$\mathcal{C}_{\mathcal{C}}$	9 2 3 3 3 3 3	100	57 x	クト 11日 11日	347	
ACTUAL A-6E FRE	TIME	0000	20.	10.0	-20	χω-	- ~	19	000	200	ρm	70	OW	N.W.	w.	20.	4-1	no	000	NOO	
	ALT	909. 1124. 990.	200	267	288	37C	851	96	966	12	986	05	967	264 109	128 152	500	983	964	2212	933	
	TAS	и и о и и и е	40	275	400	44	からっ なって	212	といっ	324	140	302	23	1 4 1	080	687	200	27 27 27	1 0 0 0	111	
:	DEG	000													-0					000	



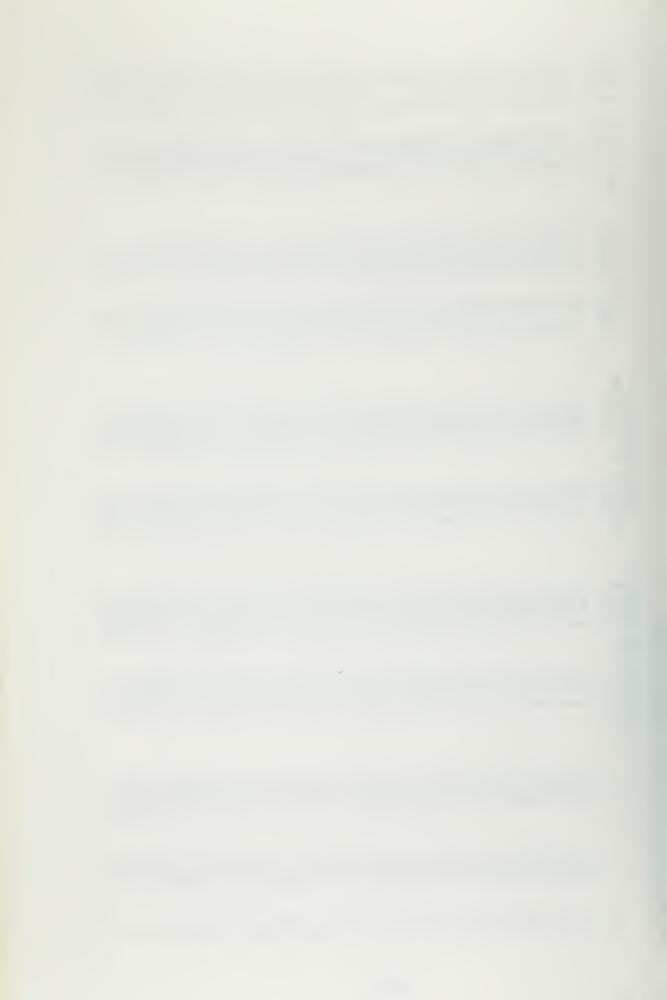
'n
ENTS
Z
ш
-
C
-
11
H
ō
COEFFIC
0
010
_
WITH
-
=
2
Ŋ
MOD-5
0
2
9
7
IK-76
X

PER CENT ERROR TIME DIST		20.937 20.937 20.4000 20.400 20.400 20.400 20.400 20.400 20.400 20.400 20.400 20.4000 20.400 20.400 20.400 20.400 20.400 20.400 20.400 20.400 20.4000 20.400 20.400 20.400 20.400 20.400 20.400 20.400 20.400 20.4000 20.400 20.400 20.400 20.400 20.400 20.400 20.400 20.400 20.4000 20.400 20.400 20.400 20.400 20.400 20.400 20.400 20.400 20.4000 20.400 20.400 20.400 20.400 20.400 20.400 20.400 20.400 20.4000 20.4000 20.400 20.400 20.400 20.400 20.400 20.400 20.400 20.400 20.4000 20.400 20.400 20.400 20.400 20.400 20.400 20.400 20.400 20.4000 20.400 20.400 20.400 20.400 20.400 20.400 20.400 20.400 20.4000 20.400 20.400 20.400 20.400 20.400 20.400 20.400 20.400 20.400 20.400 20.400 20.400 20.400 20.400 20.400 20.400 20.400 20.40
ER ENCES DIST		10105
D'I FFE TIME	00000000000000000000000000000000000000	0000
MODIFIED ALGORITHM DIST	44444440000000000000000000000000000000	00mm0 00mm0
NPS BOEING TIME		14400
DELIVERY EZE DATA DIST	4444m444mmmmm4m4m4mmmmv4444mmvmv4444m 0494mm404mmvmmmmmmmmmmmmmmmmmmmmmmmmmmmm	11370
ACTUAL A-6E FRE TIME		0.44/0
ALT	11 11 10000000000000000000000000000000	82700
TAS	のなりは、のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。のなった。<	48897 41974
DEG	000000000000000000000000000000000000000	



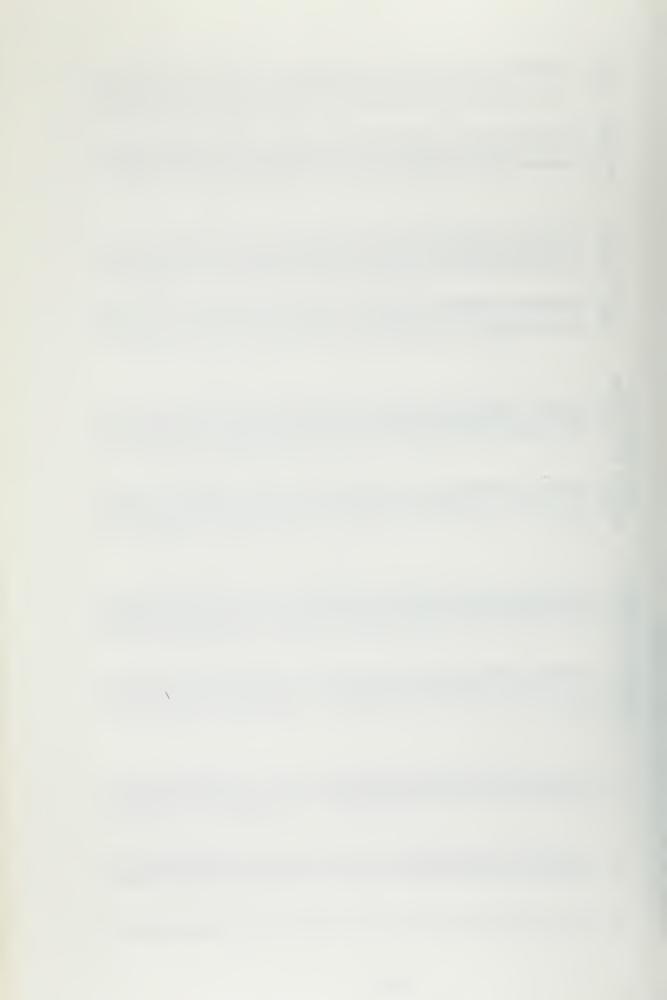
S
Z
=
ن
$\overline{\Box}$
щ
ш.
90
COEFFICIENTS
\Box
OLD
I
-
WITH
_
5
1
MOD-5
$\frac{1}{2}$
_
9
7
MK-76
Ī

PER CENT ERROR TIME DIST	
ER ENCES DIST	
DIFFE	00000000000000000000000000000000000000
MODIFIED ALGORITHM DIST	のというできないというでは、これでは、これでは、これでは、これでは、これでは、これでは、これでは、これ
NPS BOEING TIME	
DEL IVERY EZE DATA DIST	00000000000000000000000000000000000000
ACTUAL A-6E FRE TIME	00000000000000000000000000000000000000
ALT	
TAS	α
DEG	OHHHOOOHOHOOOOHHHH

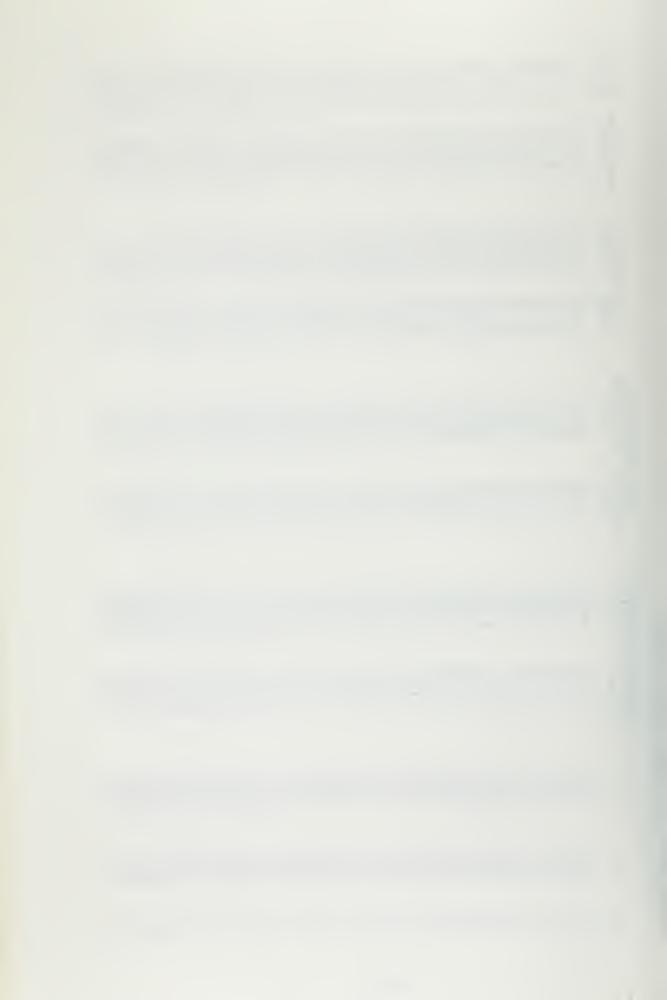


IEN
COEFFICIENTS
010
MOD-5 WITH
MK-76 MOI

	ERROR Dist	00000000000000000000000000000000000000
	PER CENT TIME	0101114724110000111172222000000000000000
	RENCES DIST	
	DIFFE	00000000000000000000000000000000000000
	MODIFIED ALGORITHM DIST	でいるようなものところところとととももももももももももものできるものののなってでした。まるころののもではなってもしまっているのとといるものでもらららららららららららららららららららららららららららららしといっているといるといっといっといっといっといっといっといっといっといっといっといっといっといっと
	BOEING TIME	
	DELIVERY EZE DATA DIST	000440mm000000000000000000000000000000
	ACTUAL A-6E FREI	100 100 100 100 100 100 100 100 100 100
	ALT	
	TAS	m m m m m m m m m m
·	DEG	



T ERROR DIST	
PER CEN	
RENCES DIST	### ### ##############################
DIFFE	## ## ## ## ## ## ## ## ## ## ## ## ##
ODIFIED ALGORITHM DIST	44466000000000000000000000000000000000
NPS M BOEING TIME	1 1 8 1 8 2 8 2 8 2 8 2 8 2 1 2 1 2 1 2
DELIVERY EZE DATA DIST	66-16-17-17-17-17-17-17-17-17-17-17-17-17-17-
ACTUAL A-6E FRE	
ALT	1000 1 1 1 1 1 1 1 1 1
TAS	muu
DEG	

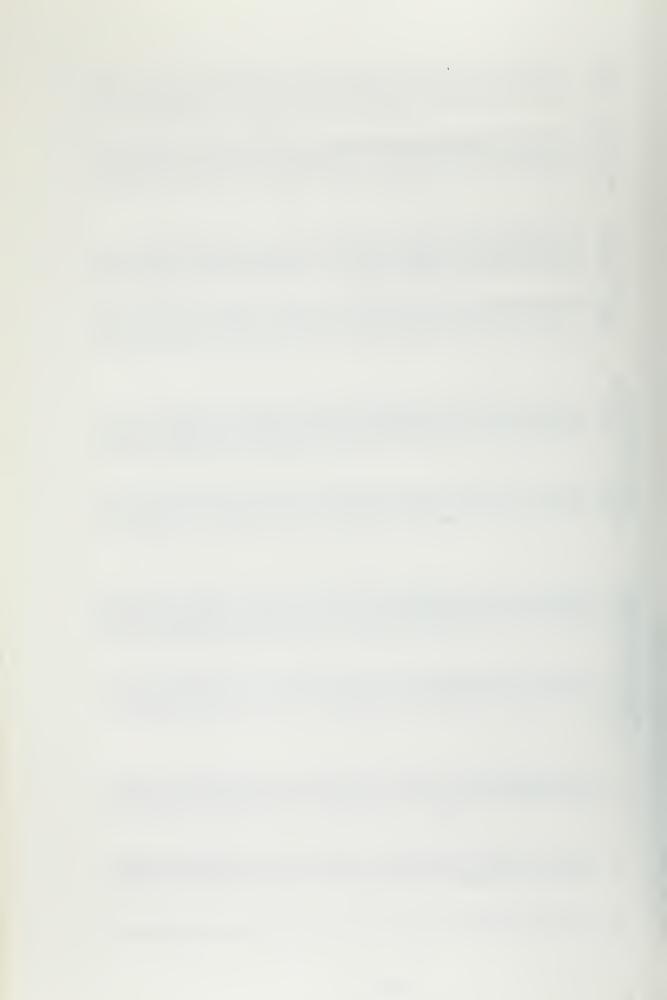


PER CENT ERROR TIME DIST		######################################
DIFFERENCES TIME DIST	00000000000000000000000000000000000000	16 - 223 06 - 223 10 - 223 10 - 223 09 - 223
S MODIFIED NG ALGORITHM E DIST	クルーの	90047568 4488941170767676767676767676767676767676767676
DATA BOEI	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	2399. 0033. 0057. 153. 359.
ACTUAL DEL		
ALT	1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	り 4 の て ろ 1 ら
TAS		0404444
DEG		



S
COEFFICIENTS
Z
ш
\vdash
ပ
\vdash
L
4
H
\sim
010
_
0
WITH
-
=
-5
M00-5
\dot{c}
0
Ĭ
_
9
4K-76
-
$\overline{\mathbf{x}}$
-

	•
PER CENT ERROR TIME DIST	
DIFFERENCES TIME DIST	00000000000000000000000000000000000000
NPS MODIFIED DEING ALGORITHM TIME DIST	7-7-7-7-7-7-7-7-7-7-7-7-7-7-7-7-7-7-7-
CTUAL DELIVERY 6E FREEZE DATA FIME	7.2000 7.2000
ALT A-	10000000000000000000000000000000000000
TAS	$\begin{array}{c} \omega_1 \omega_2 \omega_3 \omega_3 \omega_3 \omega_3 \omega_3 \omega_3 \omega_3 \omega_3 \omega_3 \omega_3$
DEG	1010000111110000011100100100100100



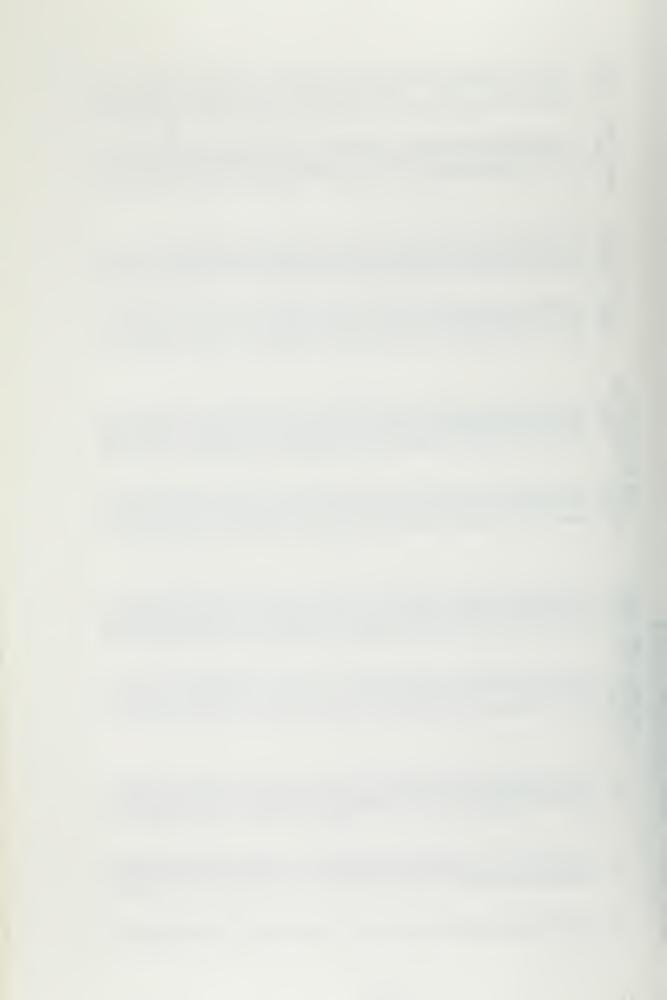
COEFFICIENTS
5
ī
-
0
$\overline{\Box}$
ŭ.
3
\ddot{c}
٩
010
Ξ
WITH
3
10
MOD-5
0
Š
_
9
4K-76
×
2

PER CENT ERROR TIME DIST	
RENCES	
DIFFE	00000000000000000000000000000000000000
MODIFIED ALGORITHM DIST	44WW7444W444W4444W44444444444444444444
NPS P BOEING TIME	88779777777777777777777777777777777777
DELIVERY EZE DATA DIST	44474444444444444444444444444444444444
ACTUAL A-6E FRE TIME	887-6-884
ALT	1006 1006 1006 1006 1006 1006 1006 1006
TAS	######################################
DEG	0-00-00-000-000-0000000000000000000000



IENTS
5
ũ
\vdash
ပ
-
i
H
COEFFIC
O
۵
010
0
_
WITH
_
2
S
MOD-5
9
¥
_
9
- 7
Ÿ
MK-76

PER CENT ERROR TIME DIST	
FF ER EN	000 001 002 005 005 005 005 005 005 005
GORITHM DI DIST	44444444444444444444444444444444444444
NPS MOD BOEING AL	111988181117201140099901171888189188918891989198918891989198919
DELIVERY EEZE DATA DIST	40744444444444444444444444444444444444
 A-6E FRI TIME	######################################
S ALT	
DEG TA	00000000000000000000000000000000000000

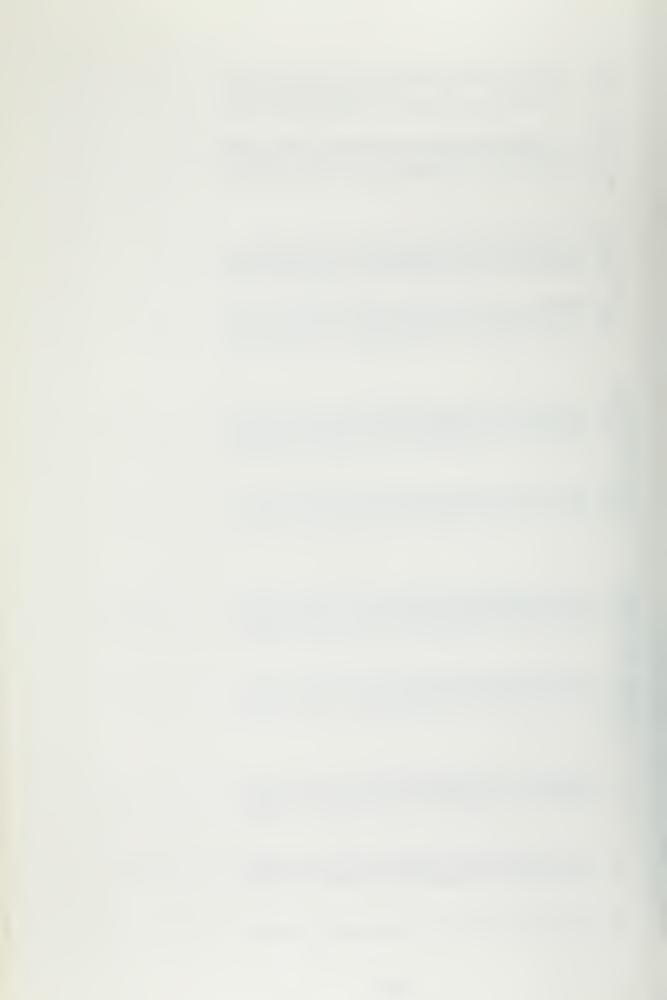


S
N N
CI
FI
COEFFICIENTS
S
010
WITH
3
MOD-5
00
1K-76
¥

PER CENT ERROR TIME DIST	1	1.01 01.00
DIFFERENCES TIME DIST		• 55
NPS MODIFIED BOEING ALGORITHM TIME DIST	7.20 7.20	070 66.
UAL DELIVERY FREEZE DATA ME DIST	20000000000000000000000000000000000000	076 04.
A-6E ALT A-6E	111	• 1170•
DEG TAS	10000000000000000000000000000000000000	•



MODIFIED G ALGORITHM DIFFE DIST	### ### ##############################
NPS BOEIN TIME	トトトトトクトトトトのトトトトの ではらするのでは、できるでは、 のいっては、 のいっては、 のいっては、 のいっとは、 のい。 のい。 のい。 のい。 のい。 のい。 のい。 のい。
CIENTS L DELIVERY REEZE DATA DIST	00000000000000000000000000000000000000
OLD COEFFICIENT ACTUAL DEL A-6E FREEZE TIME	トトトトトゥトトのアーのトーのののアートトトーク
WITH	880886448088600000000000000000000000000
<-76 MOD-5	
M O	0000011100001000110001000



APPENDIX E

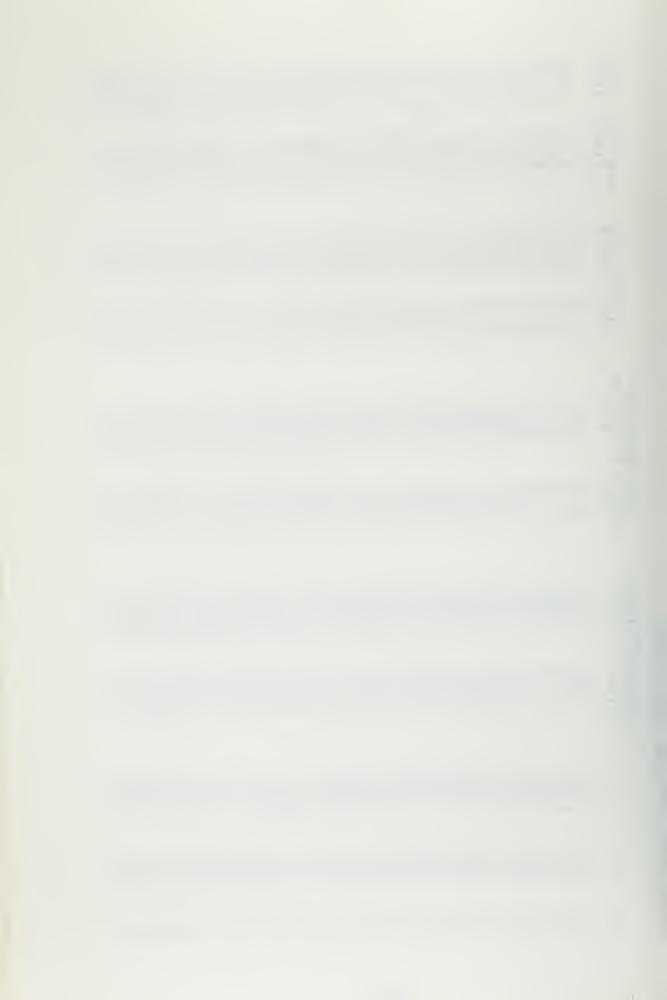
This appendix compares the experimental data with the ballistics algorithm using the new coefficients for drag, mach, and weapon type.



RENCES 01ST	
DIFFE TIME	00000000000000000000000000000000000000
MODIFIED ALGORITHM DIST	$\begin{array}{c} \sigma v \sigma \sigma v \sigma v \sigma v \sigma v \sigma v \sigma \sigma \sigma \sigma \sigma \sigma $
NPS P BOEING	1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
ENTS DELIVERY EZE DATA DIST	$\frac{1}{2}$ $\frac{1}$
COEFFICIE ACTUAL D A-6E FREE	00000000000000000000000000000000000000
WITH NEW	11111111111111111111111111111111111111
M00-5	ちちらちらららららなっているというというというというというというとしているとしているとしているとしているとしているとしているというというというというというというというというというというというといいないには、いいいいいいいいいいいいいいいいいいいいいいいいいいいいいいいいい
MK-76 DEG	0-

ERROR DIST

C EN ∃



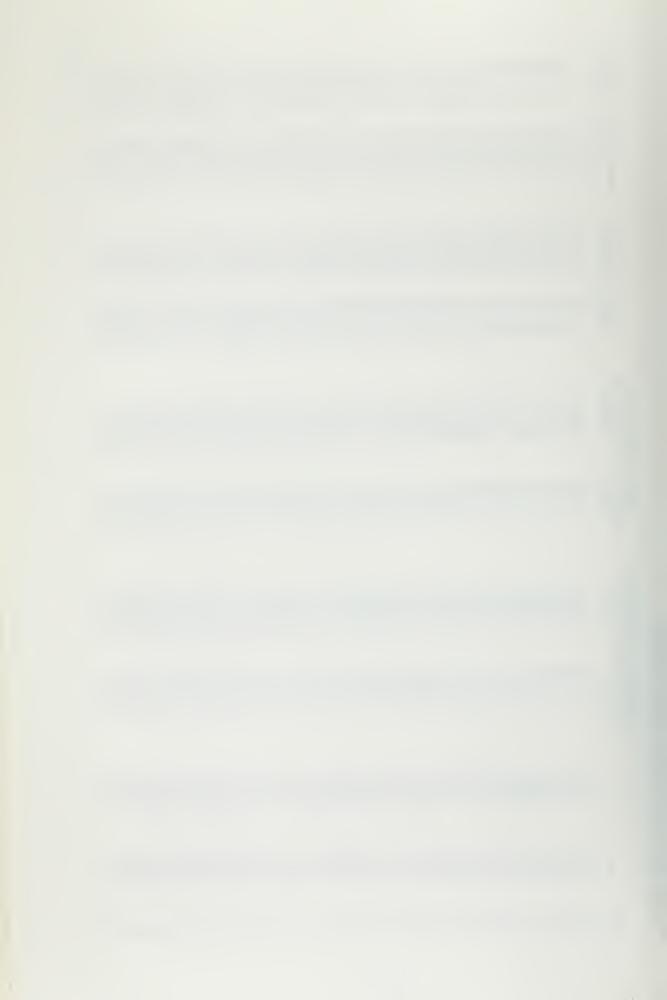
S
5
Ξ.
_
S
Ξ
ī
Щ
COEFFICIENT
_
3
ZEW
4
I
WITH
3
_
'n
M00-5
ö
Σ
0
MK-76
Ī
¥
2

PER CENT ERROR TIME DIST		
DIFFERENCES TIME DIST	00000000000000000000000000000000000000	27
OD IFIED ALGORITHM DIST	0.444444444444444444444444444444444444	•
NPS MOEING	て毎て毎日毎日でしてしまりとしまりとしまりとしましてものできるしまって、 のいっしょうかのものできるのではもなってものできるとしまってでいるとは、 のいっとは、よっなでは、これでは、これでは、これでは、これでは、これでは、これでは、これでは、これ	•
DELIVERY EZE DATA DIST	00444444444444444444444444444444444444)
ACTUAL A-6E FRE		
ALT	10000000000000000000000000000000000000	•
TAS	<i>мамамамамамамамамамамамамамамамамамама</i>	4
DEG)



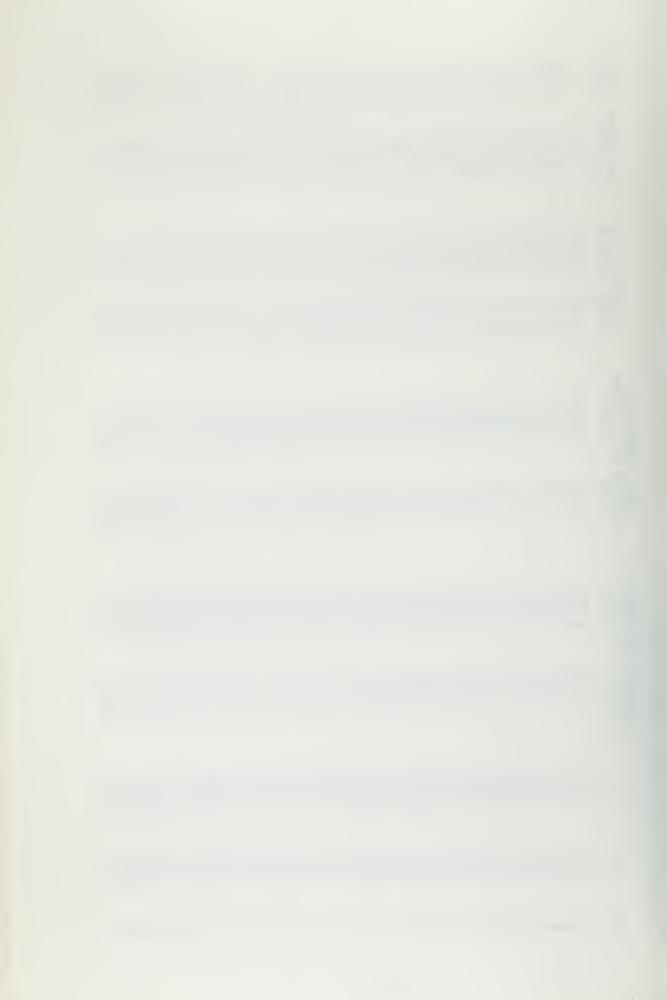
COEFFICIENTS
2
H
-
S
H
щ
щ
H
2
_
3
回
NEW
I
WITH
2
S
1
0
MOD-5
2
5
MK-76
1
×
Σ

PER CENT ERROR TIME DIST	
RENCES	
DIFFE	00000000000000000000000000000000000000
MODIFIED ALGORITHM DIST	4444444666666666666666666666666666666
NPS POEING TIME	
DELIVERY EZE DATA DIST	444#444###############################
ACTUAL A-6E FRE TIME	
AL T	11 101 1000 1000 1000 1000 1000 1000 1
TAS	まるほとろうとうとうとうとうとうとうとうとうとうとうとうとうとうとうとうとうとしょうとこれのもこととなるなったなかならでしょうというとうとうとうない。。。。。。。。。。。。。。。。。。。。。。。。。。。。
DEG	000000000000000000000000000000000000000



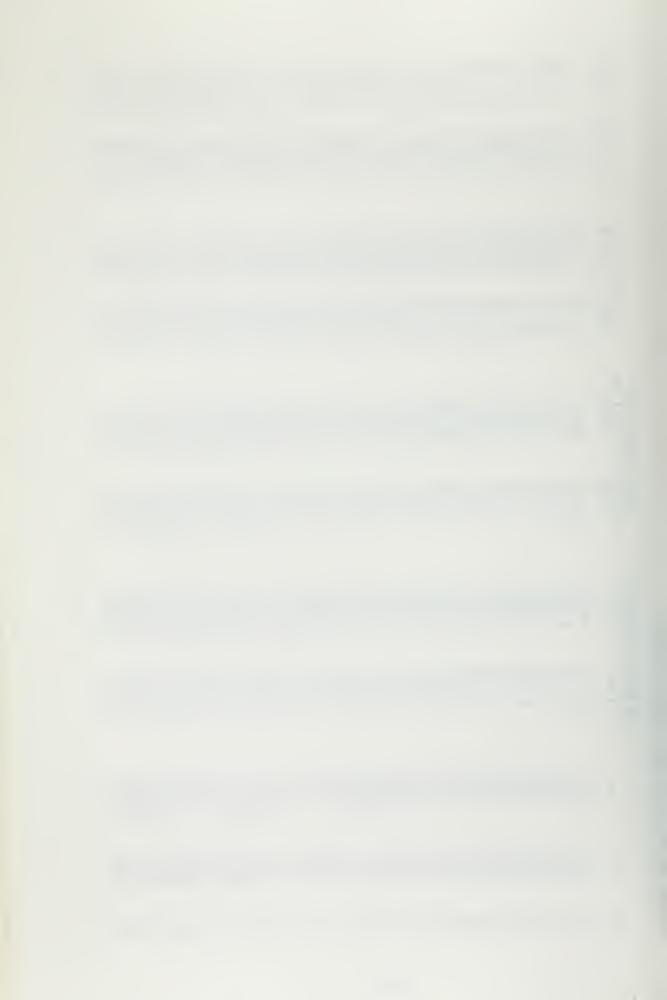
S
<u></u>
Z
3
$\ddot{\mathbf{G}}$
=
L
L
7
COEFFICIENT
NEN
<u></u>
~
I
-
WITH
2
1
\subseteq
M00-5
_
9
MK-76
J
÷
_

NT ERROR DIST	
PER CE TIME	
RENCES DIST	
DIFFE	01000000000000000000000000000000000000
ODIFIED ALGORITHM DIST	のあるとしています。 できるというでは、 できるというできます できます できまり できまり できまり できまり できまり できまり できまり できまり
NPS M BOEING TIME	11 01 00 00 10 10 10 10 10 10 10 10 10 1
DEL IVERY EZE DATA DIST	00000000000000000000000000000000000000
ACTUAL CA-6E FREE	
ALT	
TAS	α
DEG	CHAHOOOHOHOOOOOHHHHOOOOOOHHHH



TS
CIEN
COEFFICIENTS
NEW
WITH
M00-5
4K-76

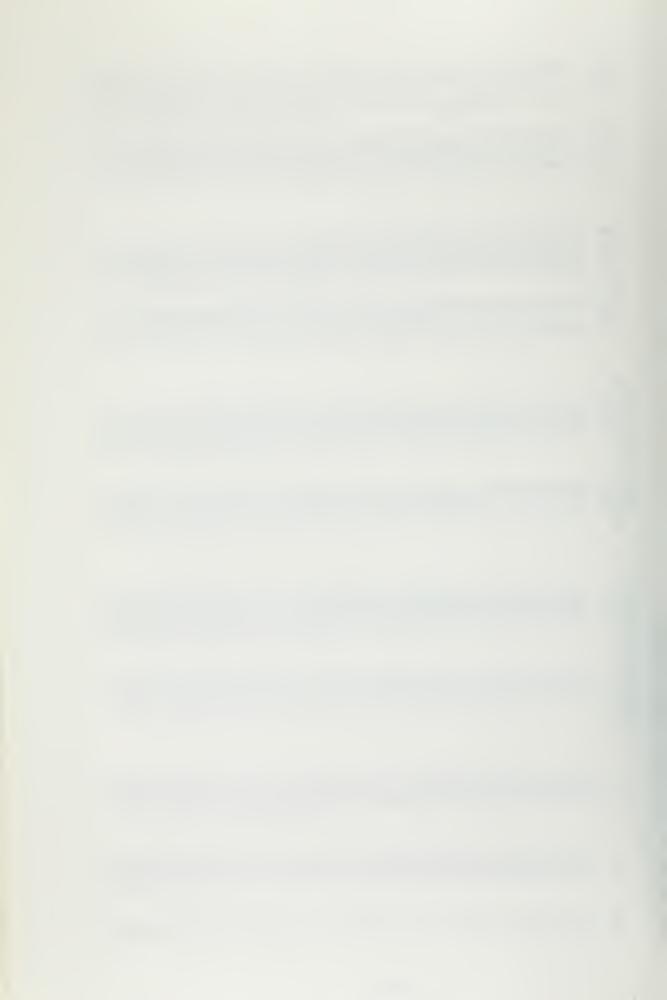
PER CENT ERROR TIME DIST	
DIFFERENCES TIME DIST	
	00000000000000000000000000000000000000
OD IFIED ALGORITHM DIST	00000000000000000000000000000000000000
NPS M BOEING TIME	1111 1000-1000-1000-1000-1000-1000-1000
OELIVERY EZE DATA OIST	44400000000000000000000000000000000000
ACTUAL A-6E FRE	100// 2000/104/4440// 2000/0000000000000000000000000
ALT	
TAS	
DEG	



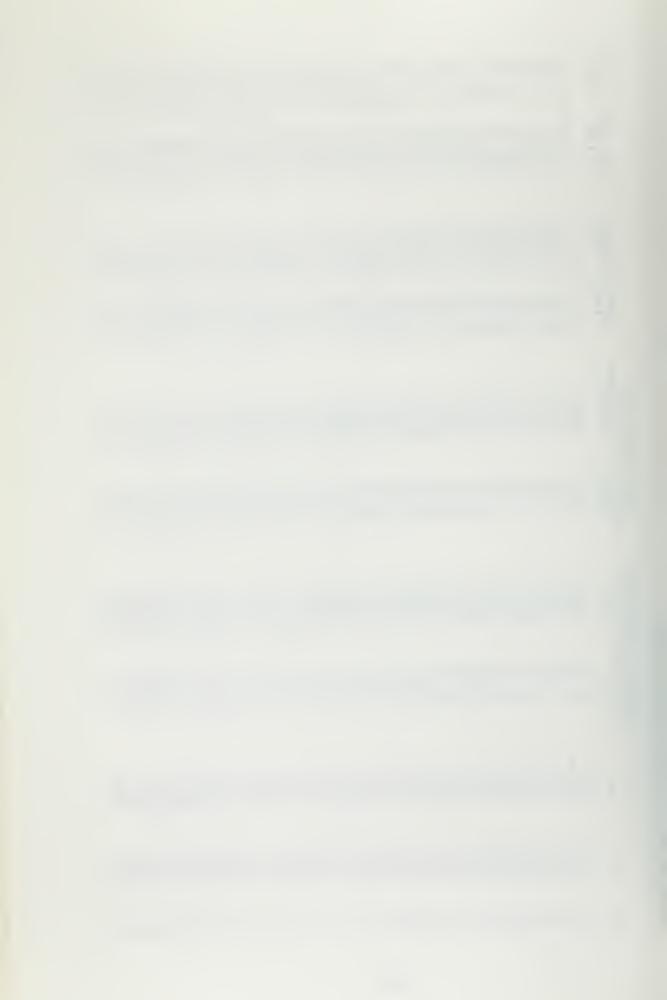


S
=
Z
ш
$\overline{}$
\subseteq
Ξ
正
ш
COEFFIC
ပ
_
NEW
=
_
I
E
3
10
41
Ċ
Ö
M30-5
K-76
-
1

PER CENT ERROR TIME DIST	
DIFFERENCES TIME DIST	00000000000000000000000000000000000000
NPS MODIFIED DEING ALGORITHM TIME	77-78-78-88-88-77-77-78-78-88-77-8-78-78
JAL DELIVERY FREEZE DATA B	85000000000000000000000000000000000000
ACTU A-6E ALT TIM	i i i i i i i i i i i i i i i i i i i
DEG TAS	

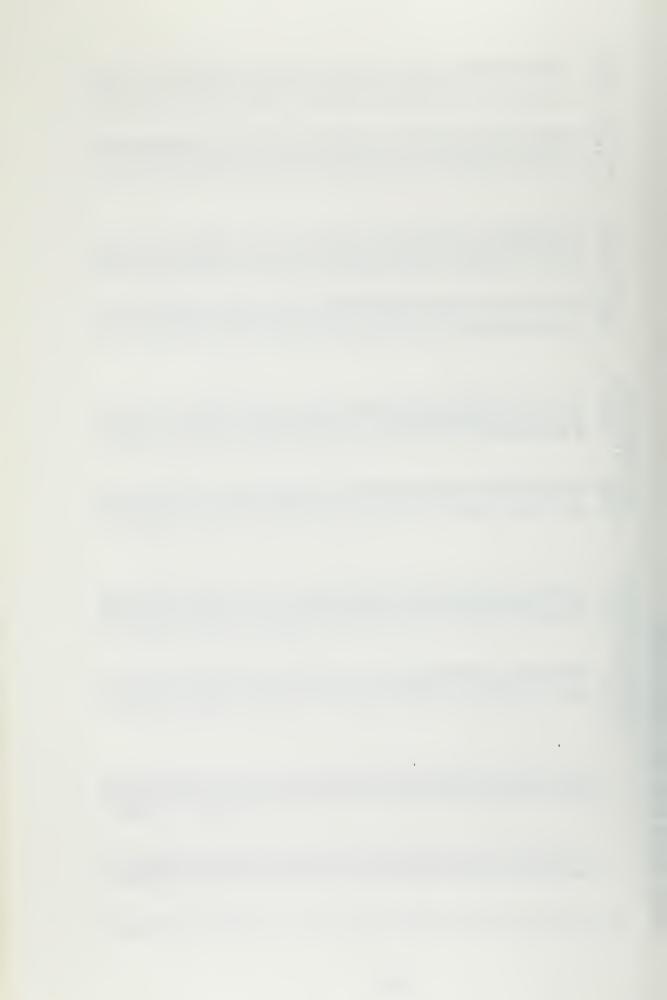


PER CENT ERROR TIME DIST	
ERENCES DIST	
DIFF	00000000000000000000000000000000000000
M <u>odified</u> Algorithm Dist	44mumu4444444mumu4amunumu44m444444mumu44mu444444mumu44mu44m
NPS BOEING TIME	トレートののファートのののののでは、いっとしてファートレートレート (10mmののののののでは、10mmののののののののののののののののののののののののののののののののののの
DELIVERY EZE DATA DIST	44744444444444444444444444444444444444
ACTUAL A-6E FREI TIME	478777677888776778888888778778 4688888877677788767888888776677788767 468888877677788767888888776677788767
ALT	10000000000000000000000000000000000000
TAS	α
DEG	



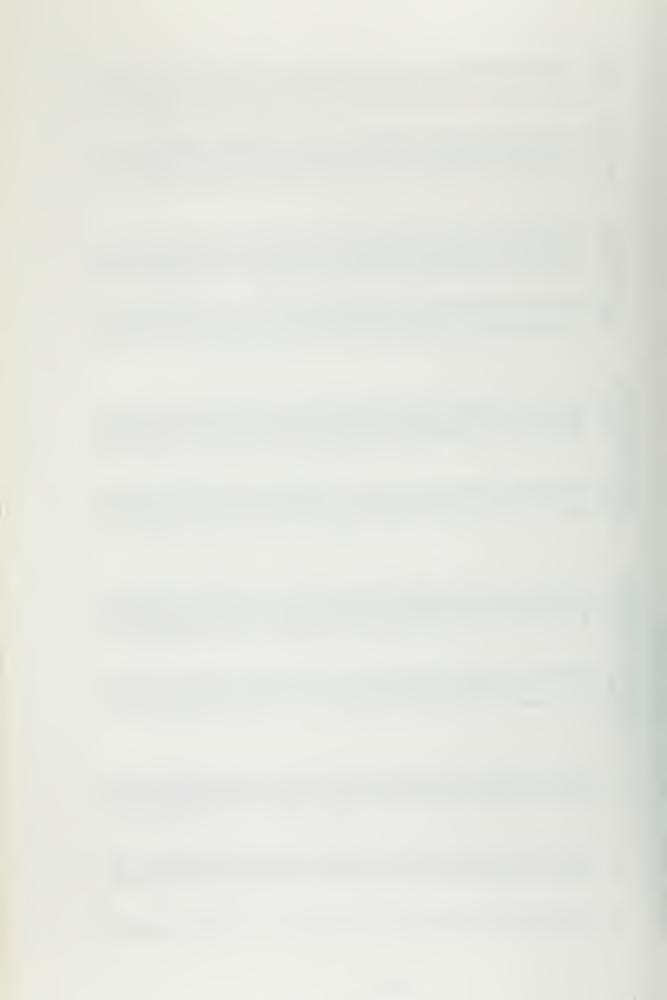
S
-
Z
COEFFICIENTS
S
\vdash
L
L
ш
0
C
NEW
ш
Z
WITH
=
_
MOD-5
–
3
\Rightarrow
2
.0
4K-76
1
-
=

	·
T ERROR DIST	1 1 1 1 1 1 1 1 1 1
P ER CEN	
RENCES DIST	1
DIFFE	00000000000000000000000000000000000000
OD IFIED ALGORITHM DIST	44 44 44 44 44 44 44 44
NPS M BOEING TIME	04000000000000000000000000000000000000
DEL IVERY EZE DATA DIST	44474444444444444444444444444444444444
ACTUAL A-6E FRE TIME	88779788977777789999999999999999999999
ALT	11 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
TAS	α
DEG	0-00-000000000000000000000000000000000



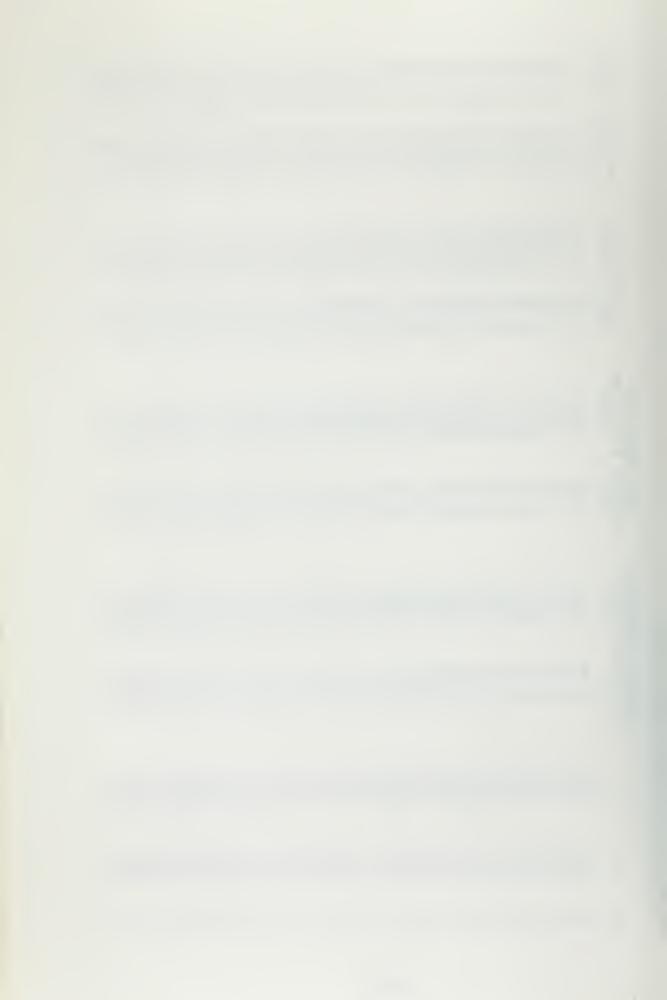
S
FICIENTS
Z
ш
-
S
\vdash
щ
ш
COEF
0
C
_
-5
N E
~
_
H
ITH
WITH
WITH
MOD-5 WITH
MOD-5
MOD-5
MOD-5

PER CENT ERROR. TIME DIST	
RENCES	
DIFFE	00000000000000000000000000000000000000
MODIFIED ALGORITHM DIST	4 日本 本本 本本 本本 本 本 本 本 本 本 本 本 本 本 本 本 本
BOEING TIME	1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
DELIVERY EZE DATA DIST	4の444444444444のようなない。 4の4444444444なるののでは、 4の44444なるののでは、 4の4444などののできませた。 4の404のでは、 4の404のようなできません。 1の5464のようなできません。 1の5464のようないできません。 1の54640のようないできません。 1の54640のようないできません。 1の54640のようないできません。 1の54640のようないできません。 1の54640のようないできません。 1の54640のようないできません。 1の54640のようないできません。 1000のないできません。 1000のないできません。 1000のないできません。 1000のないできません。 1000のないできません。 1000のないできません。 1000のないできません。 1000のないできまないできまないできまないできまないできまないできまないできまないできま
ACTUAL A-6E FREI TIME	1 10 10 10 10 10 10 10 10 10 1
ALT	89880000000000000000000000000000000000
TAS	
DEG	000000000000000000000000000000000000000



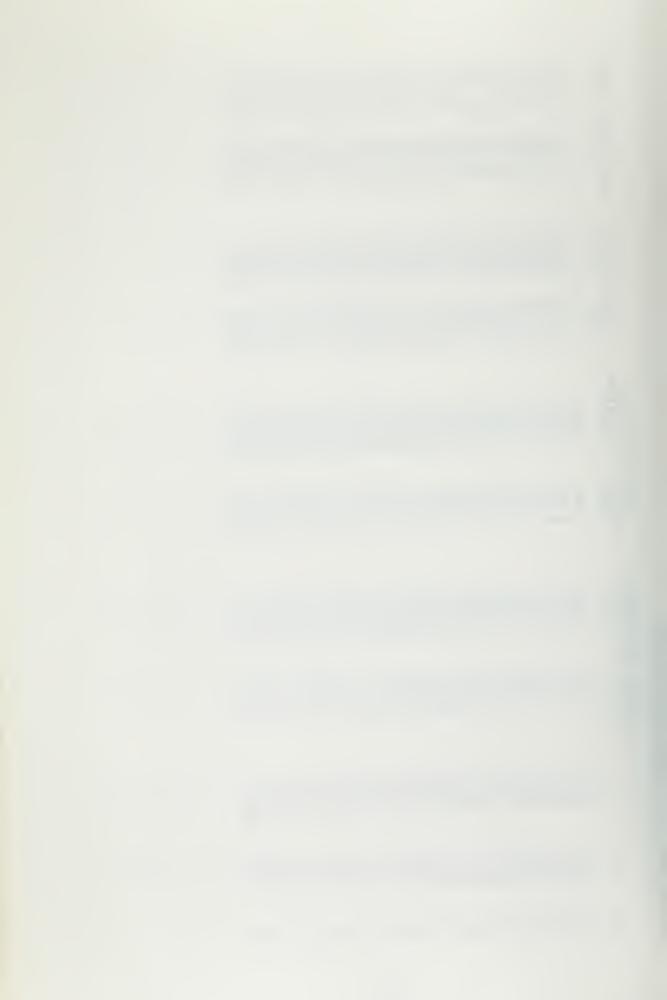
SIN
IE
C
COEFFICIENTS
00
NEW
WITH
MOD-5 1
K-76

	PER CENT ERROR TIME DIST	22000 25
	RENCES	
	DIFFE	00000000000000000000000000000000000000
	MODIFIED ALGORITHM DIST	の
	NPS POEING TIME	
)	DEL IVERY EZE DATA DIST	0.04040000000044444440004400004444000000
• • • • • • • • • • • • • • • • • • • •	ACTUAL A-6E FRE TIME	
	ALT	11000000000000000000000000000000000000
	TAS	444
	DEG	



S
1
COEFFICIENTS
=
S
-
H
ш
2
0
NEW
ž
HIIM
-
3
10
ï
0
MOD-5
~
9
MK-76
¥
Σ

PER CENT ERROR TIME DIST	11. 1. 1. 1. 1. 1. 1. 1. 1. 1. 1. 1. 1.
DIFFERENCES TIME DIST	00000000000000000000000000000000000000
NPS MODIFIED BOEING ALGORITHM TIME DIST	88877778787878888877777777777777777777
ACTUAL DELIVERY A-6E FREEZE DATA TIME DIST	7.50 7.50 7.50 7.50 7.50 7.50 7.50 7.50 7.50 7.50 7.50 8.30 7.50 8.40 7.50 7.50 8.40 7.50
ALT	100989889999877788998998998998999899999999
TAS	$ \frac{1}{2} $
DEG	000000



```
PRINT HEADING INFORMATION AND WEAPON CONSTANTS
SWITCH = 0.0
CALL IO
GC TO 30
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       CALCULATES TIME OF FALL AND DOWN RANGE TRAVEL CALL TRAJ
                                                                                                                                                                                                                                                                                                                                                                                           INITIALIZES CONSTANTS FOR PARTICULAR WEAPON CALL DECODE
                                                                                                                                                                                                                                      SWITCH = -1.0
FORTRAN VERSION OF BALLISTICS ALGORITHM
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     INITIALIZE VARIABLES FOR EACH WEAPON
SET = 1.0
CALL SETDAT
                                                                                                                                                                                                                                                                                                                        INITIALIZES ALL WEAPON CONSTANTS
SET = 0.0
CALL SETDAT
                                                                                                                                                                                                                                                                         LASTID .EQ. IDNO) GO TO 20
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          CCMPUTES ERROR AND STATISTICS CALL STATS
                                                                                                                                                      TIALIZE PROGRAM CONSTANTS
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            PRINT RESULTS AND STATISTICS
SWITCH = 1.0
CALL IO
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          30
```



```
FOR IDNO ',12,//,
DKG1 = ',F9.7,5X,'DM1 = ',F9.7,5X,
                                                          , F9.7, 5X, 0M2 = ', F9.7, 5X,
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 PRINTS OUT WEAPON COEFFICIENTS AND HEADING INFORMATION WRITE (6,10) HOND, CFORMI, DKG1, DM1, VMUZ, DS, CFORM2, DKG2, DM2, FN, SL ITYPE, IREF, VE, IBOTH, DMAX, DTI
                                                                                                                                                                                                                                                                                                                               IF (SWITCH) 40, 50, 60
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               ပပ
                                                                                                                                                                                                                                                                                                                                                                                                                                     UU
```



```
SET ONLY ONCE AT BEGINNING OF EXECUTION
20
                    ပပ
```



```
C WEAPON CONSTANTS FOR THE MK 57 UNRETARDED

DKG1= 6.2994E-03

DT1 = 3.

GC TO 31

C WEAPON CONSTANTS FOR THE MK 61 UNRETARDED

DKG1= 4.01E-03

GC TO 31

C WEAPON CONSTANTS FOR THE MK 61 UNRETARDED

DKG1= 4.01E-03

GC TO 31
WEAPON CONSTANTS FOR THE MK 116 WETEYE 4 IREF= 2
```



					J. F.		
	φ				ELEC		
	FIREBOMB	FUZE)ZE	T Z	FUZE	FUZE
	I I	ĭ	ЭЕО	Ŧ			
907	4	MECI	TARI	MEC	CONICAL	MECH	ELEC
	AND	FIN MECH	UNRETARDED	FIN MECH FUZE		NIT	FIN
HIT	1,2,				1 3		
0		CONICAL	SNAKEYE	CONICAL	0	CONICAL	CONICAL
MOD	77 MOD			00	MOD		
76		81	81	82	82	83	83
¥	Σ ¥	Σ	Σ X	ξ	¥	Σ ¥	Σ ¥
THE)3	T HE	THE	THE	7.HE	THE	1 HE	THE
FOR TH	:0R	WEAPON CONSTANTS FOR T IREF= 1 CFORM1= 2.5704 GO TO 31	F0R	FOR	FOR	FOR	FOR
3.0 3.9077E-3.48E-03	STANTS F(4.2.00021266	570	NSTANTS FOR = 3.0 9.767E-03	4TS .064	ANTS 1.493	.343	
STAN B. 3.0 3.1	\$ TAN 2.02 1.02	STAN 31.	3 - 1 - 1 - 1 - 1 - 1 - 1 - 1 - 1 - 1 -	STAN 12.23	STAN 1.1.	STAN	STAN
TI III	X = X = X = X = X = X = X = X = X = X =	RA I	X = X = X = X = X = X = X = X = X = X =	RM1=1	RM1	RAIL TO	FE
OHOOOO NAXAXHO MAOOH	ODDIO GODDIO GAXHO MARAHO	SOLOR SOLOR SOLOR	0-000 0-000	OLUNA BOT DO GO I	OLUCA BOLUCA HOLUCA	SOLO	IRE
WEAPON CONSTANTS F 5 IREF= 2 5 DMAX = 3.0 CFORM1= 3.9077 0 KG1= 6.3648E- DTI = 1.	WEAPON CONSTANTS F 6 IREF= 4 DMAX = 2 DKG1= 0+021266 DTI = 1 GC TO 31	WEAF	WEAPON CONSTANTS I B IREF= 4 DMAX = 3.0 DKG1= 9.767E-(DTI = 2.0 GO TO 31	WEAPON CONSTANTS FOR 9 IREF= 1 CFORM1= 2.064 DTI = 3.6G TO 31	WEAPON CONSTANTS F 10 IREF= 1 CFORM1= 1.4932 DTI = 3. GC TO 31	WEAPON CONSTANTS F 11 IREF= 1 CFCRM1= 1.3431 DTI = 1.	WEAPON CONSTANTS
0 0	ာပ် (ین د) <u> </u>)))	رار



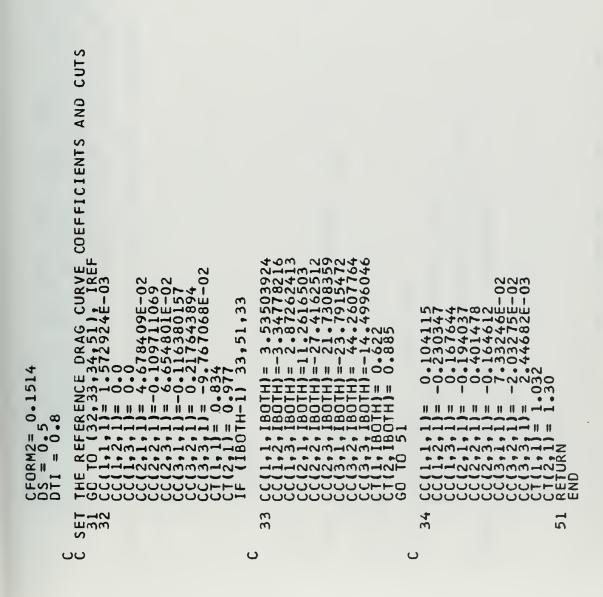
```
WEAPON CONSTANTS FOR THE MK 84 CONICAL FIN MECH AND ELEC FUZE 13 IREF= 1 CFORM1= 1.0 DTI = 3.60 TO 31
                                                                        C WEAPON CONSTANTS FOR THE MK 82 SNAKEYE UNRETARDED
17 IREF= 4
DMAX = 3.
DKG1= 0.007329
OTI = 1.
GC TO 31
                                                                                                                                                                                                                                                                                                                                                                                       C WEAPON CONSTANTS FOR THE MK 82 SNAKEYE RETARDED
                                                                                                                                               C WEAPON CONSTANTS FOR THE MK 86 WET SAND FILLED
15 IREF= 1
0 MAX = 3.
CFORM1= 3.4972
0 TI = 2.
GC TO 31
                                                                                                                                                                                                                                        WEAPON CONSTANTS FOR THE MK 88 WET SAND FILLED
16 IREF=1
CFORM1= 1.605
DTI = 3.60 TO 31
```





		SEC DELAY	SEC DELAY	SEC DELAY	
		0.4	0.8	9.0	
		RETARDED	RETARDED	RETARDED	4
	S	RETA	RETA	RETA	MOD
7	ROCKET	43	51	19	106
e n		Σ Σ	¥	Σ X	Σ̈́
五五	1 HE	1 HE	THE	THE	THE
9964 14992	FOR	FOR	FOR	FOR	FOR 14
WEAPON CONSTAN 23 IREF= 3 DMAX = 1.5 CFORM1= 2.0 DKG1= -0.0 VMUZ= 3300 DTI = 0.5	C WEAPON CONSTANTS 24 IREF= 3 ITYPE= 2 CFORM1= 0.82 CFORM2= 1.0 FN= 1746.0 DS = 1.4225 GC TO 31	C WEAPON CONSTANTS 25 IREF= 4 11YPE= 0 05= 0.98 0KG2= 1.48 0TI = 0.31 6C TO 31	C WEAPON CONSTANTS 26 IREF= 4 1 IYPE= 0 0	C WEAPON CONSTANTS 27 IREF= 4 ITYPE= 0 DS= 0.89 DKG2= 2.70 DT = 0.1 GO TO 31	C WEAPON CONSTANTS F 28 IREF= 2 ITYPE= 2 CFCRM1= 0.1514







```
THE STEP SIZE TO THE VACUMN DROP TIME REMAINING
)= DTV
                                                                                                                                                                             ((iono.Le.17).OR. (IDNO.Eq.23)) GO TO
                                                                                                               RUNGE KUTTA SUBROUTINE
CALL RUNGE
DIV = 1/G*(VY+SQRT(VY**2+2.*G*(Y)))
D = DII
TF ((IDNO:15.17), OP. (IDNO:50.23))
                                                                                                                                                                                                       HE SECOND STAGE DRAG PARAMETERS
STG = 2
STEP SIZE FOR FIRST STAGE DRAG
D= DS+SL*U
GO TO 3
                                                                                                         KUTTA SUBROUTINE
                                               C COMPUTE STEP SIZE 2 D = DMAX
```



```
DRAG PARAMETERS FOR THE FINAL INTEGRATION STEP ((IDNO.LE.17).OR.(IDNO.EQ.23)) GO TO 6
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          C UPDATE POSITION AND VELOCITIES

Y= Y0+AD*VY
RHO= 2.37576E-03-Y*(6.87557E-08-Y*6.71618E-13)
API= AP2
API= AP2
API= AP2
AN1= AN2
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          YC= Y
VXO= VX
VYO= VY
RHO= 2.37576E-03-Y*(6.87557E-08-Y*6.71618E-13)
CALL DERIV
                                                                                                                                                                                                     C UPDATE THE TIME OF FALL AND THE DOWN RANGE TRAVEL

T = T + DTV

X = X +DTV*VX

RETURN

END
                                                                                                                                                                                                                                                                                                                                                                                                                                                                         C INITIALIZE THE VARIABLES FOR THE RUNGE KUTTA AD = A*D
                                                                                                                                      C CALL RUNGE FOR THE FINAL INTEGRATION STEP 6 CALL RUNGE DTV = 1/G*(VY+SQRT(VY**2+2.*G*(Y)))
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             C COMPUTE TIME, POSITION AND VELOCITIES
```



```
MSTG)+(CC(IREG, 2, MSTG)+CC(IREG, 3, MSTG)*CM)*CM)
                                         C DETERMINE THE REGION OF THE DRAG CURVE THAT IS APPLICABLE IF (CM-CT(1, MSTG)) 10,10,20
                                                                                                                                                                                                                                                                      BALLISTIC CALCULATION
V = SQRT(VX*VX+VY*VY)

CM = V*(8.9544E-04+3.26E-09*Y)+DM
                                                                                                                                    0 50
CM-CT(2,MSTG)) 30,30,40
```



OF TIME OF FALL AND DOWN RANGE TRAVEL TES DIFFERENCE BETWEEN SIMULATION AND BALLISTICS TABLES
T TABY
X TABX
TES PER CENT ERROR OF TIME OF FALL AND DOWN RANGE TRAVE

ပပ ပ



```
/******* PROCEDURE CALLS TO THE DISK OPERATING SYSTEM *******/
/* PROGRAM DECLARATIONS */
DECLARE BDOS LITERALLY '3FFDH',
BOOT LITERALLY '0',
LF LITERALLY '10',
CR LITERALLY '13',
TRUE LITERALLY '1',
TRUE LITERALLY '1',
IBFCB (33) BYTE INITIAL (0, INPL
OBFCB (33) BYTE INITIAL (0, OUTF
INPUT * BUFFER (128) BYTE;
OUTPUT * BUFFER (128) BYTE;
OUTPUT * BYTE INITIAL (255),
IDNO BYTE;
(DEG, ALT, VKTS, TM, X) (3) BYTE;
                                                                                                                                                                                                                                                                                                                        PROCEDURE (F.A);
DECLARE F BYTE, A ADDRESS;
GO TO BDOS;
RETURN;
END MON1;
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                PRINTCHAR: PROCEDURE(CI
DECLARE CHAR BYTE
CALL MON1(2,CHAR)
RETURN;
END PRINTCHAR;
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 PRINT: PROCEDURE (A);
DECLARE A ADDRES
CALL MON1 (9,A);
RETURN;
ENC PRINT;
                                                                                                                                                                                                                                                                                                                                                                                                                                       PROCEDURE (F. CECLARE F BYTI GO TO BDOS; END MON2;
                                                                                                                                                                                                                                                                                                                                                                                                                                       MON2:
                                                                                                                                                                                                                                                                                                                             MON1:
```



```
DISK&ERROR:
CALL CRIF;
CALL CRIP;
```



```
(.IBFCB) <> 0 THEN CALL DISK*ERROR;
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   INPUT: PROCEDURE;

DECLARE I BYTE;

IF (IDNO := GET$NEXT$BYTE) = 0 THEN CALL TERMINATE;

DO I = 0 TO 2;

DEG(I) = GET$NEXT$BYTE;

END;
                                                                                                                                                                                                                     TERMINATE: PROCEDURE;
DO WHILE OUTPTR < 127;
OUTPUT$BUFFER (OUTPTR := OUTPTR + 1) = 30H;
                                                                                                                                                                                                                                                                                                (*OUTPUT$BUFFER);
(*OBFCB) <> 0 THEN CALL DISK$ERROR;
(*OBFCB);
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 RETURN I NPUT$BUFFER (INPTR := INPTR + 1);
END GET$NEXT$BYTE;
                                                                                                                                                                                                                                                                                                                                                                     . (. PROGRAM COMPLETE
                                                                 inf ( . execution BEGINS
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             (I) = GET$NEXT$BYTE;
                                                                                                   (.08FCB);
(.18FCB);
ENC CLOSE;
```



```
/***** END OF PROCEDURE CALLS TO THE DISK OPERATING SYSTEM ****/
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    /********* MATHEMATICAL FLOATING POINT PACKAGE *********
                                                                                                                                                                                                                                                                                                                                                                                                                                                                       DO:
CALL SETDMA (.OUTPUT$BUFFER);
IF DISKWRITE (.OBFCB) <> 0 THEN CALL DISK$ERROR;
OUTPTR = 255;
                                                                                                                                                                                                      LOS IF (HEX&CHAR >= OAH) AND (HEX&CHAR <= OFH) THEN
ASCII&CHAR = HEX&CHAR + 37H;
ELSE CALL CONVERT&ERROR;
RETURN ASCII&CHAR;
ENC HEX$TO$ASCII;
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    VARIABLES GLOBAL TO THE FLOATING POINT PACKAGE */
DECLARE ZE BYTE, ZZ ADDRESS,
YE BYTE, XE BYTE;
                                                                                         HEX$TO$ASCII: PROCEDURE(HEX$CHAR) BYTE;
DECLARE (HEX$CHAR,ASCII$CHAR) BYTE;
IF (HEX$CHAR >= 00H) AND (HEX$CHAR <= 09H) THEN
ASCII$CHAR = HEX$CHAR + 30H;
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  OUTPUT$BUFFER (OUTPTR := OUTPTR + 1) = ASCIII;
OUTPUT$BUFFER (OUTPTR := OUTPTR + 1) = ASCIII;
RETURN;
END PUT$NEXT$BYTE;
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         OUTPUT: PROCEDURE;
DECLARE I BYTE;
CALL PUT$NEXT$BYTE (TM(I));
END I = 0 TO 2;
CALL PUT$NEXT$BYTE (X(I));
CALL PUT$NEXT$BYTE (X(I));
END OUTPUT;
RETURN;
END INPUT;
```



```
/* CHECK TO SEE IF NUMBERS ARE WITHIN SIGNIFICANCE RANGE */
IF RANGE > 15 THEN
DO;
/* VARIABLES NOT WITHIN SIGNIFICANCE RANGE */
IF XE > YE THEN
DO;
Z = X;
Z = X;
ETURN;
EFTURN;
END;
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 /* FORM MANTISSA */

XX = SHL(DQUBLE(X) 8) OR X(1);

YY = SHL(DQUBLE(Y) 8) OR Y(1);

YY = SHL(DQUBLE(Y) 8) OR Y(1);

If (X(2) AND 80H) = (Y(2) AND 80H) THEN SIGNSEQUAL

ELSE SIGNSEQUAL = 0;
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     /* VARIABLES ARE WITHIN RANGE OF SIGNIFICANCE */
Z(2) = Y(2);
                                                                                                                                                                                                                                                                                      ** DETERMINE DIFFERENCE IN EXPONENTS */
(E = X(2) AND 7FH;
(E = Y(2) AND 7FH;
(F XE > YE THEN RANGE = XE - YE;
(LSE RANGE = YE - XE;
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      Z(1) = Y(1);
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     /* EXPONENTS EQUAL */
IF XE = YE THEN
DO;
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      Z = Y;
RETURN;
END;
                                                                                                    ZE = ZE
END;
ENC ADJUST;
                                                                                                                                                                        PROCEDURE
DECLARE ()
                                                                                                                                                                         ADD:
```



```
Z(2) = X(2) + 1;
                                                       /* X > Y */
IF YY < XX THEN
DO;
ZE = X(2);
IF SIGNSEQUAL THEN GO TO EXITI;
GO TO EXIT3;
END;
Y(2);
SIGNSEQUAL THEN GO TO EXITI;
FO EXIT2;
                                                                                                                                                                                                                                                                                 DO;

ZE = X(2);

YY = SHR(YY, RANGE);

IF SIGNSEQUAL THEN GO TO EXITI;

GO TO EXIT3;

END;
                                                                                                                                                                                                                                                                                                                                                     EXPONENT OF Y > EXPONENT OF X */
= Y(2);
= SHR(XX,RANGE);
SIGNSEQUAL THEN GO TO EXITI;
TO EXIT2;
                                                                                                                                                                                                               Z(2) = 0;
                                                                                                                                                                                                                                                       /* EXPONENT OF X > EXPONENT OF Y */
IF XE_> YE THEN
                                                                                                                                                                     Z = X; Z(1) = X(1);
RETURN;
END;
                                                                                                                                               /* X = Y */
IF SIGNSEQUAL THEN
DO:
```



```
ZE;
                                                                                                                                                                                                                                                                                                                                                                           Z(2) =
                                                                                   Iŧ
                                                                                 2(2)
                                          /* WHEN THE SIGNS ARE DIFFERENT AND Y > X EXIT2:
                                                                                                             ×
                  Z(1) = LOW(ZZ);
                                                                                                                                                                                TRACTION ROUTINE */
A BYTE:
BYTE:
                                                                                                     /* WHEN SIGNS ARE DIFFERENT AND X > Y EXIT3:
                                                                                 Z(1) = LOW(ZZ);
                                                                                                                                                Z(1) = LOW(ZZ);
                                                                                                                                                                                                                                                   ) = YY(1);

) = YY(2) XOR 80H;

(XA, YYMINUS, ZA);
ZE = ZE + 1;
END;
Z = HIGH(ZZ); Z
RETURN;
                                                               ZZ = YY - XX;
CALL ADJUST;
Z = HIGH(ZZ);
RETURN;
                                                                                                                             ZZ = XX - YY
CALL ADJUST;
Z = HIGH(ZZ)
RETURN;
ENC ADD;
                                                                                                                                                                                                                                        YYMINUS =
YYMINUS(1)
YYMINUS(2)
CALL ADD (
RETURN;
ENC SUB;
                                                                                                                                                                                    * FLOATING PC
DECLARE (X
```



```
Y(2) AND 7FH) -40H + ZE)
                                                                                                                                                                                                                                                                                                                                                                                            (XSIGNS EQUAL */
(XSIGN = YSIGN) THEN
DO:
XE = X(2) AND
                                                                                                                                                                                                                                                                                                                                                                      XSIGN = X(2) AND 80H;
YSIGN = Y(2) AND 80H;
= ZZ + XX
```



```
/* SIGNS POSITIVE */
IF XSIGN = 0 THEN */
IF XSIGN = 0 THEN */
IF XSIGN = 0 THEN RETURN 0;
IF XE < YE THEN RETURN 0;
IF XX > YY THEN RETURN 0;
IF X > OH THEN BYTE;
IF X = OH THEN BYTE;
IF X = OH THEN BYTE;
IF X = OH THEN BOH) XOR (Y(2) AND 80 XE = X(2) AND 7FH;
IF X = X(2) AND 7FH;
XE =
```



```
Z(2) = SGN OR O7FH;
                                                                                                                                                                                                                                                                             DO;
Z = 0;
END;
END;
Z = HIGH(ZZ);
END DIV;
END DIV;
                                                                                                 JÜÄŘEROOT ROUTINE */
IS POSITIVE REAL NUMBER */
ADDRESS;
XA BYTE;
                                                                                                                                                                                                                                                                  Z(11) = 0FFH;
      ZE = ZE + 1;
                                    IF YY > XX THEN YY = SHR(YY,1);
ELSE ZE = ZE + 1;
                                                                                                                                                                                                            /* OVERFLOW/UNDERFLOW */
EXIT:
IF ZE > 7FH THEN
DO:
IF XE > YE THEN
DO:
Z = OFFH; Z
                                                                 2Z = 0H; DO I = 1 TO 16; TEMP = XX - YY;
DO;
2Z = 8000H;
GO TO EXIT;
END;
```



```
COS DATA (OFFH,80H,40H,0FBH,83H,40H,0F3H,9AH,40H,
0E7H,0E3H,40H,0D8H,8FH,40H,0C5H,0D9H,40H,0B0H,0CH,40H,
97H,81H,40H,0F9H,2FH,3FH,0BFH,7AH,3FH,82H,0C8H,3FH,
88H,17H,3EH,87H,0EDH,3AH),
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   ORD DATA (80H,00H,3DH,0C0H,00H,3EH,0A0H,0OH,3FH,0E0H,00H,0H,9H,9OH,0H,0H,0H,0H,0OH,4OH,0DH,4OH,0DH,4OH,0DH,4OH,0OH,4OH,0OH,4OH,0OH,4OH,0OH,4IH,0A8H,0OH,4IH,0B8H,0OH,4IH,0C8H,0OH,4IH),
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   DATA (OFFH,0D5H,3CH,0BEH,0E1H,3EH,9DH,69H,3FH,
                                                                                                                                                             40H;
                                                                                      - 40H;
APPROXIMATION OF ROOT IS */
                                                                                                                                                          IF ZE < 80H THEN T(2) = SHR(ZE,1)
ELSE T(2) = 40H - SHR(-ZE,1);
                                                                                                                                                                                                                                                                                                                                                                                                                    H
                                                                                                                                                                                                              ZZ = SHL(DQUBLE(X), 8) OR X(1);

ZZ = SHR(ZZ,1) OR 8000H;

T = HIGH(ZZ);

T(1) = LOW(ZZ);
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  COS$SIN: PROCEDURE (THA, MAGA);

/* FLOATING POINT COSINE AND SINE
/* 0.0 <= THETA <= PI/2 */
DECLARE (THA, MAGA) ADDRESS,
N BYTE,
N BYTE,
THETA BASED THA BYTE,
THETA BASED THA BYTE,
THETA BASED THA BYTE,
                                                                                                                                                                                                                                                                                                                                                                                                                2(2)
BYTE,
                                BYTE;
BYTE;
 ZA
BASED
BYTE,
                                                                                                                                                                                                                                                                                                                                                                                                                    H
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 NIS
                                                                                                                                                                                                                                                                                                         CCALL
CCALL
NCALL
NOD
                                                                                                                                                                                                                                                                                                                                                                                                                Z = T; Z
RETURN;
END SQRT;
 NABH
```



```
40H,0A2H,76F,40H,0B9H,0DCH,40H,
,40H,0EDH,6DH,40H,0F7H,82H,40H,
H,40H);
 008H, 0ECH, 3FH, 88H, 0CEH, 5BH, 40H, 0DFH, 0FH
```

```
THETA(2)
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      X(2)=41H;
Y(2)=00H;
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       E,
C9H,10H,43H);
(OC9H,10H,0C1H);
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         DO WHILE THETA(2) > 80H;
CALL ADD (.THETA,.PITWO,.THETA);
END;
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 DO WHILE THETA(2) > 43H;
CALL SUB (.THETA,.PITWO,.THETA);
END;
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 TRIG: PROCEDURE (XA,YA,THA);

/* FLOATING POINT TRIGNOMETRY ROUTINE */
DECLARE (XA,YA,THA) ADDRESS,
X BASED XA BYTE,
Y BASED YA BYTE,
TH BASED THA BYTE,
THETA (3) BYTE,
THETA (3) BYTE,
THETA (3) BYTE,
THETA (3) BYTE,
THETA (4) BYTE,
THETA (4) BYTE,
THETA (5) BYTE,
THETA (6) BYTE,
THETA (6) BYTE,
THETA (6) BYTE,
THETA (7) BYTE,
THETA
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               (THA, ORD(N), DIF);

(DIF, COS(N), TEMP);

TEMP(2) 1;

(TEMP, SIN(N), TEMP);

(TEMP, DIF, TEMP);

(COS(N), TEMP, MAGA);
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              THETA(1) = TH(1);
THETA(2) > 3DH THEN
D0;
ZE = 8 - (THETA(2)-3DH);
N = 3 * SHR(THETA, ZE);
END;
N = 0;
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      X(1)=00H;
Y(1)=00H;
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       . 0 THEN
DO;
X=80H;
Y=00H;
RETURN;
END;
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          11
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               CALL SUB (TEMP(2) = 1
CALL ADD (CALL ADD (CALL ADD (CALL SUB (CALL
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       IF THETA
                                                                                                                                                                                                                                                                                                                                                                                                                             EL SE
```



```
TEMP(2) = THETA(2);
                        ĎO WHILE THETA(2) < 80H;
TEMP = THETA; TEMP(1) = THETA(1);
I = I + 1;
CALL ADD ( THETA, MPIHALF, THETA);
END; /* WHILE */
                                                                                                                                                                           THETA(2) = THETA(2) AND 7FH;
                                                                                                                                                                                                                                                                                                                                                                                                                 CASE *
```



RATION STATEMENTS FOR THE SETDAT PROCEDURE*/
5.RAD.AI,AA.YT,VYK,FRACT) (3) BYTE;
052,CFORMI,CFORM2,DMI,DM2,DKG1,DKG2,VMUZ,VE,SL,FN, DMAX) (3) BYTE;
17YPE,IBOTH,J.SET) BYTE;
J,DEL,TEMPIX,TEMP2X,TEMP3,TEMP4,TEMP5,TEMP6,TEMP7,V,THETA,VXA,VYA)
8 PYTE;
18 BYTE;
19 BYTE;
10 S) (3) BYTE;
11,DS) (3) BYTE;
11,DS) (4) BYTE;
11,DS) (6) BYTE;
11,DS) (7) BYTE;
11,DS) (8) BYTE;
11,DS) (9) BYTE;
11,DS) (18) BYTE; (21) BYTE 1,082H,046H,08EH,0FAH,03BH,0B3H,033H,040H,0B6H,0DBH,040H, 1,000H,000H,0A0H,000H,0C3H,080H,000H,040H); 2 (6) BYTE 1,000H,000H,0A0H,0A0H,043H); 3 (2) BYTE /******* END OF MATHEMATICAL FLOATING POINT PACKAGE ********* / *********************** BALLISTICS PROGRAM **************** TION STATEMENTS FOR THE DECODE PROCEDURE*/

[481) BYTE:
[1,05] (3) BYTE:
[1,18] BYT 6); END; GNED IN THE DECODE PROCEDURE OFH,043H); S TO THEIR SELECTED VALUES J=0 TO 2 BY 1;



```
43,156,169,182,195,208,221,
4551);
for */
                                                                APPROX TO SQRT (U**2+VE**2)
                                                                                                                                                                                                                                                                                                                                                                                                              HREE, FOUR) LABEL;
POSITION OF THE VARIOUS WPN COEFF */
U= VKTS(IV)*1.6878
                                                                                                                                                                                                                                                BOTH THE X AND Y DIRECTIONS
                                                                                                                                              THETA= DEG(IANG) *RAD
                                                                                                                                                                               IT TO 2PI */
DO;
                                                               V=U+1/2*(VE**2)/U
```



```
H,04CH,0DAH,040H,04BH);
MATRIX OF DRAG COEFFICIENTS
```



```
* WEAPON CONSTANTS FOR THE CONTROLL ON THE CON
                                                                                                                                                                                                                                                               *
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       */
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              */
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            */
                                                       */
                                                                                                                                                                                                                                                                                                                                                                                                         *
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   *
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              *
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        *
                                                                                                                           *
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          *
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     *
```



```
AND THEN BRANCH ACCORDINGLY */
; ELSE DO;
THE VARIABLES FOR DUAL STAGE WPNS
DNO-1));
HEIR RESPECTIVE VALUES FROM THE WPNCODE
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           V= SQRT(VX*VX+VY*VY)
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    TEMP3C, TEMP4C, TEMP5C) (3) BYTE;
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   6H,0E0H,006H,024H);
ABEL:
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           WEAPON
                                                                                                                                                                                                                                                                                                                                                                                                                                             TWO: 000
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                THREE:
```



```
EIGHT: "IF (2 = COMPARE GOOFT CONTROLLY OF THE STANDARY OF THE
                                                                                                                                       CM= V*(8.955E-04+3.26E-09*Y)+DM
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  EMP2C, TEMP3C);

C, TEMP1C, TEMP4C);

EMP4C, TEMP5C);

C, DM, CM);

GION OF THE DRAG CURVE WHICH IS APPLICABLE */
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             (2 = COMPARE(.CM,.CT(MSTG+3))) THEN GO TO NINE;
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              $5 (9) BYTE
B2H,038H,093H,0A6H,029H,08DH,00BH,018H)
IE AD VALUE AD= A*D */
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             S THEIR INITIAL VALUES
. TEMP2C, . TEMP3C);
                                                                 OF THE WEAPON
```



```
RHD=2.37E-03-Y*(6.87E-08-Y*6.71E-13)
                                                                                                                                                                                                                                                                                                                                                                                                                                                                           RHO=2.37E-03-Y*(6.87E-08-Y*6.71E-13)
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 EMP38, TEMP48);
8, TEMP48, TEMP58);
EMP58, TEMP68);
B, TEMP68, RHO);
CALL TO THE DERIV PROCEDURE
                                                                                                        MP2B, TEMP4B);
MP2B, TEMP4B, TEMP5B);
*TEMP5B, TEMP6B);
MP1B, TEMP6B, RHQ);
T CALL TO THE DERIV PROCEDURE
                                                                                                      CALL MULI(.Y.TEMP2B.TEMP6B);
CALL SUB(.TEMP5B.TEMP6B);
CALL SUB(.TEMP1B.TEMP6B.RHQ);

* MAKE THE FIRST CALL TO THE DERIV PROCEDURE
LL DERIV;
* UPCATE THE POSITIONS AND THE VELOCITIES
/* Y= YO+AD*VY */
/* Y= YO+AD*VY */
/* Y= YO+AD*VY */
/* Y= YO+AD*VY */
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              CCMPUTE THE TIME, POSITION AND VELOCITIES
                                                                                                                                                                                                                                                                                                                                                                                                                                 TEMP28);
DENSITY
                                                                                                                                                                                                                                                                                                                                                                                                                       *
```



```
TRAJECTORY PROCEDURE
```



```
)= TEMP2A(1); TEMP3A(2)= TEMP2A(2)+1;
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         i)= TEMP2A(1); TEMP3A(2)= TEMP2A(2)+1;
4A);
4A);
4A; TEMP5A);
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            DTV= 1/6*(VY+SQRT(VY**2+2.*6*Y))
                                                                                                                                                                                                                                                                                                                                                                                   HE DRAG PARAMETERS FOR THE FINAL INTEGRATION STEP */

S= 6; TABLE1= 27;

(ITYPE=2) THEN DO; MSTG= 0; TABLE1=0; END;

J=0 TO 2 BY 1;

S(J)= DKG2(J);
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              RAVEL OF THE WEAPON X= X+DTV*VX
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    EMPTA);
[V);
OF THE WEAPON TM= TM+DTV */
                                                                                                                                                                                                  EN DO; MSTG=0; TABLE1= 0; END;
                                                                                                                                                                                                                                                                                                                                                                                          **CALL RUNGE FOR THE FINAL INTEGRATION */
CALL MULT(-60, 7);

CALL MULT(-60, 7);

TEMP3A= TEMP2A; TEMP2A);

CALL MULT(-60, 7);

CALL MULT(-60, 7);

TEMP3A= TEMP2A; TEMP3A(1)=

CALL MULT(-00, 7);

CALL MULT(-00, 7);

CALL MULT(-00, 7);

CALL MULT(-00, 7);

CALL ADD(-TEMP4A; TEMP)

ALL ADD(-TEMP4A; TEMP)
                                                                             54, TEMP64);
1, 0(1, 0) = DTI(1);
60 TO SEVEN;
60 TO SEVEN;
766 DRAG PARAMETERS */
```



```
/************** END OF BALLISTICS PROGRAM ************
RETURN;
END TRAJ;
                                                                                                                               EOF
```



LIST OF REFERENCES

- 1. Grumman Aerospace Corporation, Navy Model A-6E Aircraft, Integrated Weapon System Theory, NAVAIR 01-85ADF-2-10.1, September, 1973.
- de Linhares, T. P. B., <u>Distributed Microcomputer Air-borne Tactical System</u>, M.S. Thesis, Naval Post-graduate School, Monterey, California, December 1975.
- 3. Jupin, J. A., <u>The Ballistics Processor of a Multiple Processor Airborne Tactical System</u>, M.S. Thesis, Naval Postgraduate School, Monterey, California, June 1975.
- 4. McCracken, W. L., <u>Design Study of an Avionics Navigation</u>
 <u>Microcomputer</u>, A.E. Thesis, Naval Postgraduate
 School, Monterey, California, June 1974.
- 5. <u>Tactical Manual Ballistics Tables</u>, NAVAIR 01-1C-1T-1, October 1973.



INITIAL DISTRIBUTION LIST

		No.	Copies
1.	Defense Documentation Center Cameron Station Alexandria, Virginia 22314		2
2.	Library, Code 0212 Naval Postgraduate School Monterey, California 93940		2
3.	Department Chairman, Code 52 Department of Computer Science Naval Postgraduate School Monterey, California 93940		1
4.	Associate Professor Uno R. Kodres, Code 52Kr Department of Computer Science Naval Postgraduate School Monterey, California 93940		1
5.	LT John T. Ertlschweiger, USN 10424 Greenford Drive San Diego California 92126		1







CHIVAN 24414 Thesis E654 Ertlschweiger 167849 c.1 Verification and feasibility study of a micro-computer based ballistics algorithm. 1944477 24414 167843 Thesis Ertischweiger E654 Verification and c.1 feasibility study of a micro-computer based ballistics algorithm.

thesE654
Verification and feasibility study of a

3 2768 002 06223 4
DUDLEY KNOX LIBRARY